SSW8

8th ISCA Speech Synthesis Workshop

August 31st - September 2nd, 2013
Barcelona, Spain

PROGRAM AND PROCEEDINGS

Arxiu de l'Institut d'Estudis Catalans. Foto: Santi Muxach

UNIVERSITAT POLITÈCNICA DE CATALUNYA BARCELONATECH

8th ISCA Workshop on Speech Synthesis

Proceedings

Barcelona • August 31 – September 2, 2013

Edited by Antonio Bonafonte

At the time of release, the proceedings can be downloaded from the website of the SSW8: ssw8.talp.cat

Contents

Message from the Chair

Welcome to the eight edition of the ISCA Speech Synthesis Workshop. Twenty-three years have passed since the first edition, in Autrans, France, 1990. It is a considerable period of time, and we have seen several generations of systems: synthesis by rule, diphone concatenation, unit selection synthesis and, last years, parametric/statistical speech synthesis. Each generation implies significant improvements on aspects as naturalness, intelligibility, flexibility, range of application or robustness. However, we have to recognize that many of the objectives of a decade ago, have still not been reached. For instance, while synthetic image and synthetic music is in some cases preferred to the real ones by the entertainment industry, synthetic speech is still just a defective copy of human speech. Human voices include a lot of information which we still are not able to code in the synthetic voices.

Each speech generation synthesis system has started by some innovative pioneer works followed by an explosion of research contributions that consolidate, extend and exploit the new technology;and finally, a stabilization where a lot of effort produces limited improvements. From my point of view, speech synthesis using Hidden Markov models is reaching this stabilization phase. At that phase, new revolutionary ideas are needed to significantly push the technology beyond of today limits. While there are still many contributions to be done in this field of parametric statistical synthesis, we need open minds to explore new approaches to speech synthesis that may become new paradigms.

The workshop program includes three invited conferences that we thing can be inspiring talks. On the first day, Dr. Heiga Zen will present recent application of deep learning to statistical parametric speech synthesis. In the last years, it has been proved that deep neural networks are better than traditional Gaussian Mixtures for acoustic Modeling in Speech Recognition. Several papers have already been published applying the same ideas to speech synthesis. This seems a promising direction. On the second day, Prof. Nigel Ward will give a talk about prosodic patterns in dialog. It is widely accepted that we need to include information of higher level (as semantic/pragmatic) to produce conversational engines. Prof. N. Ward has analyzed the prosody patterns on real dialogs from several perspectives. Finally, on Monday 2nd, Prof. Xavier Serra will explain the evolution of synthetic singing voices and the research of his lab on this field. Synthetic Music has achieved very high quality and can be used to create synthetic songs as natural and expressive as songs produced by real singers. Although synthetic singing is a close area, traditionally these are developed by two different research communities and this is an opportunity to learn their approach and their perspectives to this topic.

The technical program includes 51 regular papers. Each paper has been reviewed by three reviewers. The program is composed of 6 oral sessions for a total of 20 papers and 3 poster sessions for 31 papers. Furthermore, there are 4 additional presentations in a demo session. I am very grateful to all the contributors for their interest and efforts that have made possible to organize the workshop.

The Blizzard Challenge 2013 Workshop, September 3, is organized by Simon King, Alan W Black, Keiichi Tokuda and Kishore Prahallad. The contribution of these evaluation campaigns to the progress of the area is remarkable. It is a pleasure that this edition can be hosted by Universitat Politècnica de Catalunya.

During the preparation of the workshop I have collaborated with many people and I am sincerely grateful to all of them. First of all, I want to mention the advisory committee for their helpful advises. I would like to thank all members of the Program Commitee for their excellent work of reviewing the papers in a very tight schedule. It has been a pleasure to work with the program co-chairs, Daniel Erro and David Escudero.

I am most grateful to Olga Nuñez and Yolanda López, from the Technical Secretariat, for all the support organizing the workshop and for their constant work.

Welcome to SSW8! I am looking forward to an exciting workshop and rapid progress in the field. I wish you a wonderful workshop time in Barcelona.

Barcelona, August 2013
Antonio Bonafonte, Workshop Chair

Organizing Committee

Chair:
Antonio Bonafonte, Universitat Politècnica de Catalunya

Members:
Daniel Erro, Ikerbasque – University of the Basque Country
David Escudero, Universidad de Valladolid
Asunción Moreno, Universitat Politècnica de Catalunya
Jordi Adell, PAL Robotics (Barcelona)
Ignasi Esquerra, Universitat Politècnica de Catalunya
Francesc Alías, La Salle – Universitat Ramon Llull

Advisory Committee

Gerard Bailly, CNRS/INPG, France.
Alan Black, CMU, USA.
Nick Campbell, Univ. of Dublin, Ireland.
Rolf Carlson, KTH, Sweden.
Thierry Dutoit, Faculté Polytechnique de Mons, Belgium.
Wolfgang Hess, Univ. of Bonn, Germany.
Julia Hirschberg, Columbia Univ., USA.
Simon King, Edinburgh University, UK.
Bernd Möbius, Saarland University, Germany.
Jan van Santen, OHSU, USA.
Juergen Schroeter, AT&T Labs, USA.
Paul Taylor, Google, UK.
Keiichi Tokuda, Nagoya Institute of Technology, Japan.

Program Committee

Program Chairs:
Antonio Bonafonte, Universitat Politècnica de Catalunya, Spain
Daniel Erro, Ikerbasque – University of the Basque Country, Spain
David Escudero, Universidad de Valladolid, Spain

Members & board of reviewers:
Jordi Adell, Pal-Robotics S.L., Barcelona, Spain
Francesc Alías-Pujol, La Salle – Universitat Ramon Llull, Barcelona, Spain
Gerard Bailly, GIPSA-Lab, Grenoble, France
Roberto Barra-Chicote, Universidad Politécnica de Madrid, Spain
Jerome Bellegarda, Apple Inc., USA
Alan Black, Carniege Mellon University, Pittsburgh, USA
Andrew Breen, Nuance Communications, UK
Nick Campbell, University of Dublin, Ireland
Rolf Carlson, KTH, Stockholm, Sweden
Alistair Conkie, AT&T Labs, NJ, USA
Ricardo Cordoba, Universidad Politécnica de Madrid, Spain
Christophe D'Alessandro, CNRS-LIMSI, Orsay, France
Thierry Dutoit, TCTS Lab., Numediart Institute, University of Mons,Belgium
Raul Fernandez, IBM Research, Yorktown Heights, NY, USA
Juan-María Garrido, Universitat Pompeu Fabra, Barcelona, Spain
Xavi Gonzalvo, Google, UK
Inma Hernáez, University of the Basque Country, Spain
Wolfgang Hess, IfK Universität Bonn, Germany
Julia Hirschberg, Columbia University, NY, USA
Ignasi Iriondo, La Salle – Universitat Ramon Llull, Barcelona, Spain
Simon King, University of Edinburgh, UK
Esther Klabbers, Oregon Health & Science University, USA
Javier Latorre, Toshiba Research Europe, UK
Zhenhua Ling, University of Science and Technology of China (USTC)
Bernd Moebius, Saarland University, Germany
Juan Montero, Universidad Politécnica de Madrid, Spain
Asunción Moreno, Universitat Politècnica de Catalunya, Spain

Eva Navas, University of the Basque Country, Spain
Kishore Prahallad, International Institute of Information Technology
Eduardo Rodríguez Banga, University of Vigo, Spain
Juergen Schroeter, AT&T Labs, NJ, USA
Joan Claudi Socoró, La Salle – Universitat Ramon Llull, Barcelona, Spain
Frank Soong, Microsoft Research Asia, China
David Suendermann, DHBW Stuttgart, Germany
Jianhua Tao, Chinese Academy of Sciences, Beijing
Paul Taylor, Google, UK
Tomoki Toda, Nara Institute of Science and Technology, Japan
Keiichi Tokuda, Nagoya Institute of Technology, Japan
Jan Van Santen, Oregon Health & Science University
Junichi Yamagishi, University of Edinburgh, UK
Heiga Zen, Google, UK

Contributing Reviewers:
Enrico Bocchieri, AT&T Labs, NJ, USA
Kei Hashimoto, Nagoya Institute of Technology, Japan
Andrej Ljolje, AT&T Labs, NJ, USA
Taniya Mishra, AT&T Labs, NJ, USA
Hansjörg Mixdorff, BTH Berlin University of Applied Sciences, Germany
Keiichiro Oura, Nagoya Institute of Technology, Japan

Workshop Program at a Glance

Saturday, August 31

08:00	Registration opens
09:10 – 09:20	Opening
09:20 – 11:00	Oral Session 1: Prosody and pausing
11:00 – 12:40	Poster Session 1 & Coffee
12:45 – 14:45	Lunch Break
14:45 – 16:00	Oral Session 2: Open Challenges in speech synthesis
16:00 – 16:30	Coffee Break
16:30 – 17:20	Keynote Session 1: Deep Learning in Speech Synthesis
17:20 – 17:30	SynSIG Message
17.30 – 18.30	"Jam" Music Session
18.30 – 20.00	Reception at Institut de Estudis Catalans (SSW8 Venue)

Sunday, September 1

08:00	Registration opens
09:00 – 10:15	Oral Session 3: Robustness in synthetic speech
10:15 – 10:45	Coffee Break
10:45 – 12:25	Oral Session 4: Issues in HMM-based speech synthesis
12:30 – 14:00	Lunch Break
14:00 – 14:50	Keynote Session 2: Prosodic Patterns in Dialog
14:50 – 16:30	Poster Session 2 & Coffee
16:30 – 20:00	Guided visit to Sagrada Familia & Park Güell
20:00 – 22:00	Dinner at the Restaurant of the Royal Barcelona Maritim Club

Monday, September 2

08:00	Registration opens
09:00 – 09:50	Keynote Session 3: Singing voice synthesis
09:50 – 10:40	Oral Session 5: Synthetic singing voices
10:40 – 11:10	Coffee Break
11:10 – 12:50	Oral Session 6: Expressive speech synthesis
12:50 – 14:20	Lunch Break
14:20 – 15:20	Demo Session
15:20 – 17:00	Poster Session 3 & Coffee
17:00 – 17:10	Closing

Saturday, August 31

Oral Session 1: Prosody and pausing.

Saturday, August 31, 9:20 – 11:00

Chair: Alan Black

Poster Session 1

Saturday, August 31, 11:00 – 12:40

Chair: Eduardo Rodríguez Banga

Oral Session 2: Open Challenges in speech synthesis.

Saturday, August 31, 14:45 – 16:00

Chair: Simon King

Keynote Session 1

Saturday, August 31, 16:30 – 17:20

Chair: Keiichi Tokuda

Sunday, September 1

Oral Session 3: Robustness in synthetic speech.

Sunday, September 1, 9:00 – 10:15

Chair: Frank Soong

Oral Session 4: Issues in HMM-based speech synthesis.

Sunday, September 1, 10:45 – 12:25

Chair: Tomoki Toda

Keynote Session 2

Sunday, September 1, 14:00 – 14:50

Chair: Nick Campbell

Poster Session 2

Sunday, September 1, 14:50 – 16:30

Chair: Junichi Yamagishi

Monday, September 2

Keynote Session 3

Monday, September 2, 9:00 – 9:50

Chair: Asunción Moreno

Oral Session 5: Synthetic singing voices.

Monday, September 2, 9:50 – 10:40

Chair: Xavier Serra

Oral Session 6: Expressive speech synthesis.

Monday, September 2, 11:10 – 12:50

Chair: Paul Taylor

Demo Session

Monday, September 2, 14:20 – 15:20

Chair: Javier Latorre

Poster Session 3

Monday, September 2, 15:20 – 17:00

Chair: Keikichi Hirose

Automatic detection of inhalation breath pauses for improved pause modelling in HMM-TTS

Norbert Braunschweiler, Langzhou Chen

Toshiba Research Europe Ltd., Speech Technology Group, Cambridge, United Kingdom
{norbert.braunschweiler,langzhou.chen}@crl.toshiba.co.uk

Abstract

The presence of inhalation breaths in speech pauses has recently attracted more attention especially since the focus of speech synthesis research has shifted to prosodic aspects beyond a single sentence, as, for instance in the synthesis of audiobooks. Inhalation breath pauses are usually not an issue in traditional speech synthesis corpora because they typically use single sentences of limited length and therefore pauses including inhalation breaths rarely occur or they are deliberately avoided during recording. However, in readings of large coherent texts like audiobooks, there are often inhalation breaths, particularly in publicly available audiobooks. These inhalation breaths are relevant for the modelling of pauses in audiobook synthesis and can cause a reduction in naturalness when un-modelled. Therefore this paper presents a method to automatically classify pauses into one of four classes (silent pause, inhalation breath pause, noisy pause, no pause) for improved pause modelling in HMM-TTS.

Index Terms: inhalation breaths, pauses, speech synthesis, HMM-TTS, classification

1. Introduction

Inhalation breaths in speech pauses have not attracted much attention in the history of speech synthesis research. Exceptions are, for instance [1] who showed that the inclusion of inhalation breaths before an utterance improved the recall of synthesized sentences and [2] who indicated that their inclusion in a limited domain synthesizer improved its naturalness.

One of the reasons for this lack of interest is certainly its relatively small impact on quality and naturalness. Another reason is undoubtedly the limitations of traditional TTS training material which is typically limited to single sentences recorded in controlled conditions. In such a recording style the presence of inhalation breaths was typically avoided to achieve a homogenous representation of speech pauses as silent pauses. However, with the move to study prosodic aspects beyond the single sentence and in particular in the domain of audiobook synthesis inhalation breath noises are becoming more important.

One issue is the presence of pauses including inhalation breath noises in the training data. If un-modelled they can degrade synthesis quality, because often silent pauses and pauses including inhalation breath noises or other articulatory noises are considered as a single unit.

The reason for the presence of inhalation breaths in the training material comes with the move to use speech corpora which include prosodic features beyond a single sentence. Audiobooks are ideal for this because they are typically based on a coherent text and include large amounts of speech from a single or multiple speaker(s). However, unless inhalation breaths and other articulatory noises within pauses are deliberately removed, audiobooks include them. This has been observed particularly in the domain of publicly available audiobooks, as used, for instance in the Blizzard Challenge 2012 [3], but also recently studied on a French audiobook [4].

To use publicly available audiobooks as training material for speech synthesis purposes the speech first needs to be segmented and aligned with its corresponding text. This task was provided for the Blizzard Challenge 2012 by the lightly supervised sentence alignment and selection approach [5]. This approach automatically selects individual sentences from an audiobook for which it detects a match between recognition and expected text. The output of the lightly supervised approach is a set of sentences for which there is an audio file and a corresponding text file. Sentence sized audio files are cut out from the larger, typically chapter sized audio files. Cutting points are the mid points of any pause between sentences or the end of the sentence final phone if no pause is present. As such, the sentence sized audio files often include parts of inhalation breaths, articulatory noises (lip smacks, clicks, etc.) or other background noise in their leading/trailing parts as well as in their sentence internal pauses.

Many current synthesis frameworks are still very much limited to work on single sentences only. When the intervals of audio labelled as leading/trailing silence or sentence internal pause are not as clean as in the (controlled) studio recordings it can adversely affect speech synthesis quality. The synthesised leading and/or trailing silences can include inhalation breaths or other noises or a mixture of these and the same can happen in sentence internal pauses.

To avoid this problem and to provide the basis for modelling inhalation breath pauses in synthesis, an approach is presented which classifies any automatically labelled pause into one of four categories (silent pause = *pau*, inhalation breath pause = *paub*, noisy pause = *paun*, no pause = *no_pau*). The classification result can then be used during training and synthesis.

This paper is structured as follows: first the results of a pilot study are presented which used natural speech to test the impact of inhalation breath pauses on perceived naturalness. Then a new approach to automatically classify pauses into four subclasses is presented and evaluated. This is followed by the application of the classification approach to a publicly available audiobook. Based on the classification results an HMM-TTS model is trained and compared with a system using a single pause model. The the discussion addresses related issues and possible future research directions. Finally the conclusion summarizes the crucial findings.

2. Pilot study: Impact of breath pauses on naturalness

To test the influence of inhalation breath pauses on perceived naturalness a pilot study was conducted using natural speech from a German TTS speech corpus. This corpus, read by a female voice talent, had a number of longer sentences as well as sentences originally recorded as part of paragraphs and therefore included some inhalation breaths.

A standard preference listening test was conducted contrasting presence vs. absence of inhalation breath pauses. The test was conducted via crowd sourcing using the CrowdFlower website and subjects in Germany. The analysis included automatic cheat detection [6] and standard paired t-test for statistical significance calculation. The test used 40 sentences each of them including at least one inhalation breath pause. These evaluation sentences were selected from mixed text genres like news, navigation, audiobook, etc. and were relatively long sentences. The average number of words in the evaluation sentences was 30.3 ± 14.0. There was a total of 174 pauses (79% *paub*, 20% *pau*, 1% *paun*) and the average number of pauses was 4.3 ± 2.4 with a mean pause duration of 354.9 ± 155.3 ms.

For the preference test all inhalation breath pauses and noisy pauses as well as any noisy leading/trailing silences were manually "silenced" by using wavesurfers (http://www.speech.kth.se/wavesurfer/) "Amplify" function to reduce the amplitude of any (inhalation breath) noise to almost zero, i.e. to make it in-audible and effectively create a silent pause while still not silencing it completely. Complete "silencing" was avoided because it can result in un-natural transitions from speech to pause and vice versa. The amplitude flattening was carefully applied to avoid touching any transition to and from the pause as long as the (inhalation breath) noise could be made inaudible. The effect of the amplitude flattening was checked auditorily and repeated application was conducted when there was still audible (inhalation breath) noise.

The test compared the natural sentences (henceforth "Natural") against the sentences with amplitude flattened pauses (henceforth "No_breath"). Subjects were asked: "Indicate which of two speech sound files sounds more natural?". 660 pair stimuli were evaluated by 30 subjects. The results are presented in Table 1 and show no clear preference but a very small and statistically non-significant tendency to prefer the natural stimuli.

Table 1: *Result of preference listening test in pilot study.*

Natural	No_breath	none	p-score
35.9%	31.8%	32.3%	0.147

This result indicates that there is some impact of inhalation breath pauses on perceived naturalness, but the impact is small and statistically non-significant.

Coming back to inhalation breath pauses in speech synthesis, the first issue to address is the detection of inhalation breath pauses in the training data. The next section presents an approach to automatically classify pauses into the mentioned four sub-classes (*pau*, *paub*, *paun*, *no_pau*) based on pause labels automatically annotated by forced alignment.

3. Automatic classification of pauses

Figure 1 depicts the basic steps from a speech corpus to an acoustic model for HMM-TTS and shows the location of the pause classification method introduced below.

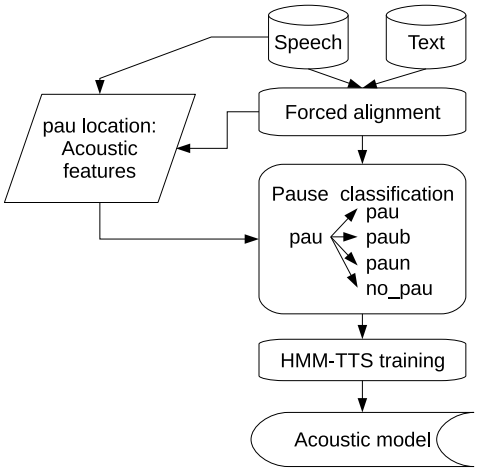


Figure 1: *Flow diagram from speech and text to acoustic model via automatic pause classification.*

Typically a TTS corpus consists of speech and corresponding text files. These are input to the forced alignment module which provides alignments of any phones and pauses. The locations of these pauses are used by the pause classifier to extract acoustic features from the corresponding stretches of speech. The output of the classifier is then used in HMM-TTS model training to train an acoustic model. The next section will give a brief explanation of the classification method.

3.1. Classification method

To enable the separation of silent pauses and pauses filled with inhalation breaths or other noise a method was developed which classifies each speech pause into one of the following four classes:

- *pau* - silent pause
- *paub* - pause including inhalation breath
- *paun* - pause including any other noise than paub
- *no_pau* - no pause

The *no_pau* class was introduced to account for the fact that some of the automatically annotated pauses are incorrect pause insertions and need to be deleted from the alignments.

The same classification is also provided for any leading and trailing silence in a given speech file with the exception that no silence will be deleted but a warning message will be printed when the classifier identifies a silence which is deemed to include speech.

The classification method uses a set of acoustic features and the type of the preceding and following phone to perform the classification. The features are scored by a set of rules and then threshold values are used for the final classification.

Feature values are extracted on a frame level and various statistics are calculated for either the whole pause or specified parts of it. The set of acoustic features is described in section 3.3. The next section presents some considerations about the input data to the classifier which are the automatically aligned phones and pauses.

3.2. Automatic alignment of pauses

Unless there is manually annotated data available, automatic phone and pause alignments are usually the basis for an HMM-TTS voice. Typically pauses are labelled automatically as part of forced phone alignments using HMM models. These models can be trained on a given corpus by a flat start approach or using speaker independent models trained on large, multi-speaker corpora. In the presented approach the automatic pause alignments were created by a flat start method which - after some iterations of maximum likelihood training - introduces a one state short pause tee-model, which is tied to the centre state of the silence model. Models are then iteratively refined and the short pause model is changed to a three state tee-model, where each state is tied to the corresponding state in the silence model.

Since this short pause model shares states with the silence model the presence of silences including inhalation breaths affects the performance of the short pause model. One issue with automatic pause alignments is the precision of the pause boundaries. Typical problems are the inclusion of parts of neighbouring sounds into the pause itself. This issue will be addressed in section 3.3 in more detail. In the presented approach no attempt was made to alter the HMM model topology. The automatic silence and pause alignments were used as provided by this method. The next section presents the acoustic features used in the classifier in more detail.

3.3. Acoustic features

At first glance the separation of silent pauses and inhalation breath pauses seems to be a straightforward approach. Silent pauses are expected to contain no voicing, very little and constantly low energy, whereas pauses filled with inhalation breaths are expected to include no voicing as well, but a higher energy level and a distinct spectral energy distribution.

To see whether these expectations can be used to classify pauses a small subset of pauses, including all four classes, were analyzed with respect to their acoustic features including f0 (for voicing detection), RMS-amplitude, and spectral energy distribution.

During this analysis it was noticed that the pause boundaries placed by the automatic aligner were not always precise but often included parts of neighbouring phones. When the preceding phone was a stop, for instance, the stop releases could be partly or completely subsumed within the pause. Depending on the type of phone before the pause the onset part of the pause could include voiced or unvoiced parts having various non-silence spectral intensities. Sometimes more than half the pause was covered by the preceding sound. Similar observations were made for the pause ends. This meant that any method using acoustic features to classify pauses had to account for these imperfectly placed pause boundaries.

Figure 2 shows schematic representations of the amplitude tracks for the three pause classes *pau*, *paub*, *paun*, indicating onset and offset parts, which can include features from neighbouring phones and which can vary according to the precision of automatically placed boundaries.

Given the analysis a set of acoustic features was chosen based on the considerations to detect presence/absence of voicing, levels of energy and distribution of spectral energy within pauses - to enable a reliable classification of each pause into one of the four classes mentioned.

The ESPS-tools *get_f0* and *sgram* were used to calculate f0, RMS-amplitude, and a series of FFT's providing information about spectral energy distribution. The *get_f0* tool was used

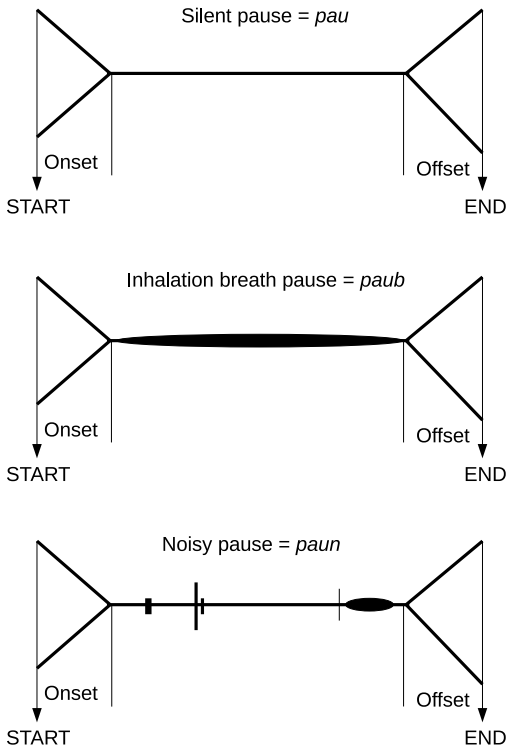


Figure 2: *Schematic representation of amplitude tracks of pause classes: pau, paub, and paun.*

with standard settings, i.e. a frame step of 10 ms. The *sgram* tool was called with the wideband spectrogram option (-m wb) which uses a frame step of 2 ms [7].

The automatically labeled start and end timestamps of each pause were used to extract these acoustic features from the corresponding stretches of each sentence sized audio file. Using automatically placed timestamps is important because the classifier is expected to work for exactly these timestamps which are not always precise as mentioned before. In case the pause duration was less than 30 ms no f0 and RMS-amplitude was calculated because the RMS value of each record is calculated on a 30 ms hanning window [7] and feature values for these were set to zero. Table 2 lists the acoustic features.

Features were introduced which observed particular parts of the pause, e.g. the first/last 20% or the first/second half. Previous internal experiments revealed that HMM-TTS synthesis quality is relatively insensitive against imprecise phone and pause boundaries which meant that imprecise pause boundaries are to some extend tolerated and should not trigger an incorrect classification or an incorrect deletion of a pause.

The features chosen include features which calculate average values across the whole pause as well as average values across defined parts of the pause. Additionally the spectral energy distribution across the full frequency range was also calculated for particular bands (low, lowmid, mid, high) based on the findings for inhalation breath pauses in the training data. Threshold values for low spectral power (perc_spec_pwr_low) and non-low spectral power (perc_spec_pwr_high) were defined, also based on observations in the training material. For all three basic features (f0, RMS, spectral energy) its mean, maximum and minimum values were calculated across the whole pause to

get an idea about levels and variances of them.

In addition to the above mentioned observations it was also noticed that a few pauses were completely incorrect, e.g. labeled within speech without any pause nearby and a few were debatable whether to be labelled as pause or being left as part of neighbouring phones especially in the case of glottal stops following or in stop - stop sequences.

Table 2: *List of acoustic features.*

Feature name	Definition
pau_dur	Duration of pause in ms
perc_voiced	% of voiced frames
f0_mean	mean f0
f0_max	max f0
f0_min	min f0
RMS_mean	mean RMS
RMS_max	max RMS
RMS_min	min RMS
RMS_std	std. dev. of RMS
spec_pwr_mean	mean spectral power
spec_pwr_std	std. dev. of spec. power
perc_spec_pwr_low	% power <10
perc_spec_pwr_high	% power >100
perc_spec_pwr_high_onset	% power >100 in 1st 20%
perc_spec_pwr_high_offset	% power >100 in last 20%
perc_spec_pwr_high_1sthalf	% power >100 in 1st half
perc_spec_pwr_high_2ndhalf	% power >100 in 2nd half
spec_bands_mean	mean all spectral bands
spec_bands_mean_std	std. dev. all spectral bands
spec_bands_max	max of all spectral bands
spec_bands_min	min of all spectral bands
spec_bands_mean_low	mean [0, 900] Hz
spec_bands_mean_lowmid	mean [1600 - 2000] Hz
spec_bands_mean_mid	mean [5000 - 6000] Hz
spec_bands_mean_high	mean [6000 - 11025] Hz

3.4. Evaluation

The next section describes the training corpus used to select the acoustic features and to define the rules in the classifier.

3.4.1. The training corpus

For training the automatic pause classifier 2291 sentences from a German speech synthesis corpus (the same as in the pilot study) were used including 4563 hand labelled pauses. This data was split into 90% training and 10% test sets. The hand labelling included the inspection of each sentence and each included pause (listening and visualising its waveform), its relabelling as either *pau*, *paub*, *paun* or its deletion. When the labeller noticed a missing pause it was inserted. Also timestamps of pauses were corrected when necessary.

The original automatic markup (henceforth called AUTO_ORIG) consisted of 2354 sentences including 4512 pauses. Because the human labeller deleted and added pauses there were 108 sentences removed in the hand markup (henceforth called HAND).

Table 3 shows how the HAND labelled pauses compare with the AUTO_ORIG pauses - for the 2354 sentences originally including a pause in the automatic markup. 75.4% are silent pauses, 15.6% include inhalation breaths, 3.9% are pauses including any other audible noise than an inhalation breath and 5% were deleted. The last row in Table 3 shows the number of pauses added by the human labeller.

Table 3: *Comparing pauses in AUTO_ORIG and HAND.*

	AUTO_ORIG	HAND
# pauses	4512	4509
pau	100%	75.4%
paub	-	15.6%
paun	-	3.9%
deleted	-	5.0%
added	-	224

The mean durations and standard deviations of the three pause classes are shown in Table 4. As can be seen the classes show distinctive mean durations: inhalation breath pauses are longest, followed by silent pauses, which have the highest standard deviation, and the noisy pauses being the shortest on average.

Table 4: *Mean pause durations and standard deviations in the training corpus.*

Pause	Mean [ms]	StdDev [ms]
pau	210.7	134.6
paub	393.1	104.2
paun	139.4	120.8

3.4.2. Results

To measure the accuracy of the pause classifier 10-fold cross-validation was conducted. For each test set it was ensured that the four pause classes were proportionally represented as in the HAND data.

Average precision was 0.87 ± 0.014 and average recall 0.86 ± 0.014. This shows that the majority of pauses were correctly classified, while on average 5.9% of *pau* labels were falsely classified as *paub*, but only 1.1% of the *paub* labels were missed. This is probably caused to a large extent by the limitations to filter out incorrect pause boundaries. When neighbouring phones are part of the pause then the acoustic features become "polluted" by these and can result in incorrect classifications. Confusions of other classes are relatively small. While the deletions are similar in numbers to the deletions in HAND, 64.3% of the pauses deleted in HAND were correctly deleted the remainder was not deleted and 18.9% were falsely deleted. This can certainly also be mostly explained by the imperfect pause boundaries which could trigger a pause deletion when the acoustic features are strongly dominated by the speech parts and not by pause parts. The classification of a pause as *no_pau* was often an indication that the automatic alignment was either completely incorrect or largely incorrect, i.e. could be used to spot mis-alignments.

To conduct another test of the classifier on unseen data it was applied to a publicly available American English audiobook. Classified pause labels were then used in the training of an HMM-TTS model and compared with the standard, single pause system. This is described in the following section.

4. Synthesis with multi-pause labels

4.1. Classifying pauses in "A Tramp Abroad"

The classifier was applied to the publicly available audiobook "A Tramp Abroad" (librivox.org) written by Mark Twain and read by John Greenman. This audiobook was part of the training material used during Blizzard Challenge 2012 (http://festvox.org/blizzard/blizzard2012.html).

For this paper the subset of sentences selected at a 100% confidence interval (by the lightly supervised sentence selection and alignment tool [5]) was used consisting of 5052 sentences and including 8624 pauses in total. These pauses were all automatically aligned by the Toshiba internal automatic phone alignment tool. From the 5052 sentences 64.7% did contain at least one pause and 36.2% did not include a pause. Without pause classification the average pause duration was 299.9 ± 155.0 ms.

After classifying each pause into the four classes of *pau*, *paub*, *paun* and *no_pau* there are 8073 remaining pauses. That means, 6.4% were deleted and the remaining ones were classified as follows: *pau*: 15.8%; *paub*: 72.6%; *paun*: 5.1%.

As presented in Table 5, the average pause durations of the three pause classes showed a similar duration pattern as in the German TTS training corpus (*paub* >*pau* >*paun*). However, the mean duration of inhalation breath pauses was shorter than in the German speaker, indicating that there are more shorter inhalation breath pauses in "A Tramp Abroad", an observation which is in line with listening impressions on several samples.

However, there are many more inhalation breath pauses in "A Tramp Abroad" than in the German TTS training corpus. This is not surprising when listening to this data which shows that the reader of "A Tramp Abroad" inhaled quite frequently, whereas the voice talent of the German speech database tried to avoid inhaling during the single sentence prompts.

Table 5: *Mean pause durations and standard deviation in "A Tramp Abroad".*

Pause	Mean [ms]	StdDev [ms]	% of original
pau	218.3	136.7	15.8%
paub	352.7	126.5	72.6%
paun	146.4	94.5	5.1%

Because there are no hand annotated pause labels for "A Tramp Abroad" it was not possible to quantify the accuracy of the classifier on this corpus. However, visual and auditory inspection of some sentences showed that the classification worked reasonably well. The next section will test the impact of multi-pause labels on synthesis quality.

4.2. HMM-TTS training

To test the impact of the classified pauses in synthesis two listening tests were conducted. An HMM-TTS voice was trained on the audiobook "A Tramp Abroad" read by John Greenman, i.e. the same audiobook as used in section 4.1 for the automatic classification of pauses.

The training corpus contained about 9 hours of speech in 4.8K utterances (the remaining sentences were left out as test set) and was sampled at 16k Hz. The acoustic feature vectors included 40 mel-cepstral coefficients, logF0, 21 band aperiodicity together with their delta and delta-delta information. They were modelled by multi-stream, 5 state, left-to-right, multi-space probability distribution hidden semi-Markov models (MSD-HSMM). The full-context HMM states were generated by introducing the phonetic, segmental, prosodic and linguistic context information. Decision tree based state clustering was used for tying the full-context HMM states based on the minimum description length (MDL) principle.

Two set of MSD-HSMMs were trained based on two phone lists. One used a single label for silence and a single label for pause, the other used multiple labels for silences and multiple labels for pauses. In the second case, the silence was divided into 3 separate "phonemes", pure silence, inhalation breath silence and noisy silence, in the same way for the short pause. In the training process, each of them was modelled individually. They were considered as different context information when full-context HMM states were generated, and the questions about the type of non-speech events were also used in the process of decision tree growing. This way, not only the different types of non-speech events were explicitly modelled, but also their influence on neighbouring phonemes was explicitly considered in decision tree generation. Thus a better distribution sharing over full-context HMM states can be achieved.

4.3. Results of listening tests

The first preference test was designed to address the question whether the finer split of pauses affects synthesis quality. 25 sentences from the test set of "A Tramp Abroad" were selected, each of them including at least one pause. To avoid the impact of automatic pause prediction, the pauses which were automatically annotated by the forced alignment were used in case of the SINGLE_PAU system and the sub-classified version of them in the MULTI_PAU system. There were 45 pauses in the automatic alignments of these sentences. These 45 pauses were classified as follows: *pau*: 4, *paub*: 38, *paun*: 2, *no_pau*: 1.

Each sentence was synthesized by two different synthesizers: synthesizer SINGLE_PAU was trained using a single pause model and synthesizer MULTI_PAU used the sub-classified pause labels for training (as described in section 4).

Subjects were asked to indicate which of two speech sound files sounds better. The test was conducted with the crowd sourcing platform CrowdFlower using subjects in the USA. 512 pair stimuli (after discarding cheats) were rated by 42 subjects. Results are presented in Table 6.

Table 6: *Results of preference listening test comparing MULTI_PAU vs. SINGLE_PAU.*

MULTI_PAU	SINGLE_PAU	none	p-score
42.6%	49.4%	8.0%	0.061

There was a statistically non-significant difference between the two systems, but a small preference for the SINGLE_PAU system. This shows that there is no significant negative impact of the multi-pause classification, but also no improvement in quality.

The second listening test was designed to address the question whether the multi-pauses add to the naturalness perception. This time paragraphs were chosen to test the impact on longer stretches of speech including multiple pauses. 20 paragraphs were selected from the test set of "A Tramp Abroad" and pause positions were taken from the original reading and not predicted in order to avoid the impact of incorrect pause prediction. The 20 paragraphs included 60 sentences which in turn included 63

pauses which were classified as follows: *paub*: 44, *paun*: 3, *pau*: 8, *no_pau*: 8.

All 20 paragraphs were synthesized by the same HMM-models trained on the multi-pause classification. However, while system MULTI_LAB used the multi-pause labels also in synthesis, system SINGLE_LAB only used the silent pauses, therefore effectively representing a synthesizer only generating silent pauses.

By listening to each stimuli it was confirmed that the SINGLE_LAB stimuli included silent pauses, an indication that the classification worked well.

Subjects were asked to indicate which of two speech sound files sounds more natural. Again the test was conducted via crowd sourcing. 328 pair stimuli (after discarding cheats) were rated by 32 subjects. The results are shown in Table 7.

Table 7: *Results of preference listening test comparing MULTI_LAB vs. SINGLE_LAB.*

MULTI_LAB	SINGLE_LAB	none	p-score
58.5%	32.0%	9.5%	$<$0.001

Subjects significantly preferred system MULTI_LAB, showing the impact of multiple pauses in training and synthesis as opposed to a single, silent pause only synthesizer.

5. Discussion

The pilot study was comparing single sentences, although relatively long ones, however, it might be necessary to use paragraphs or even longer stretches of coherent speech, because then the presence of inhalation breath pauses might be more important and appear more natural to the listener. Inhalation breaths can add to the naturalness of speech particularly in an audiobook scenario. Here, they could help to realise a more expressive reading style, when modelled correctly, for instance when particular scenes occur, where inhalation breaths help to make the story more lively, and - if omitted - might result in a lack of naturalness.

The results of the listening tests with synthetic speech showed that a more fine grained pause classification can help to improve the naturalness, especially in longer texts. However, since the test was using pause locations originally placed by the human speaker the next step that is needed, is the prediction of the more fine-grained pause classes from text. A task that is known to be difficult in the speech synthesis world, see chapter 6.2 and 6.7 in [8] about phrasing and phrasing prediction respectively.

Another aspect is the capability of the system to synthesize natural sounding inhalation breaths. While the inhalation breaths synthesized with the Toshiba HMM-TTS did sound acceptable there is certainly potential for improvement.

Furthermore this work may provide the basis for subsequent work about the patterns of pause positioning and pause duration timing in coherent texts. By extending this work to account for the amount of detail observed in natural speech including inhalation breaths and possibly other pauses as well (i.e. "filled pauses" including laughter, vocatives, etc.) it might be possible to produce more natural sounding synthesis especially when synthesising large, coherent texts as in the audiobook scenario.

A possible extension of the current classifier would be to add the functionality to adjust pause boundaries. This could be beneficial in scenarios where pause boundary precision is more important, e.g. for unit selection systems.

6. Conclusions

This study investigated inhalation breath pauses and their influence on perceived naturalness in natural as well as in synthetic speech. A pilot study using natural speech, showed a small, but statistically non-significant preference for natural speech including inhalation breaths in pauses against a version which had the inhalation breaths silenced.

Following this study, a pause classifier was developed using a set of acoustic and phonetic features to classify each automatically labelled pause into one of four classes: silent pause, inhalation breath pause, noisy pause and no pause.

The approach was trained and evaluated on a German speech synthesis corpus and showed a good accuracy, especially with respect to the detection of inhalation breath pauses. The classifier was then applied to a publicly available American English audiobook and the classification results were used to train an HMM-TTS system which was compared against an HMM-TTS system trained on the same data, but only using a single pause model. Two listening tests were conducted, the first one testing the impact of the multi-pause labels on synthesis quality. No significant difference was found between systems. The second preference test addressed the question whether the multi pause labels improve the naturalness of synthesis. This time full paragraphs were evaluated and the multi-pause label system was significantly preferred against the single pause label.

The effect of having just silent pauses in speech synthesis can be more severe when synthesizing audiobooks and can add to the perceived naturalness of synthetic speech. Adequate modelling of pauses - either silent, with inhalation breaths or any other form - is important for non-monotonous and prosodically well-structured speech synthesis.

7. Acknowledgements

The authors would like to thank the teams behind Librivox.org and Gutenberg.org for hosting voluntarily produced audiobooks and out of copyright books. We would also like to express our gratefulness to John Greenman, who made his narration of "A Tramp Abroad" publicly accessible.

8. References

[1] D. H. Whalen, C. E. Hoequist, and S. M. Sheffert, "The effects of breath sounds on the perception of synthetic speech," in *J. Acoust. Soc. Am. Volume 97, Issue 5*, 1995, pp. 3147–3153.

[2] S. Sundaram and S. Narayanan, "Spoken language synthesis: experiments in synthesis of spontaneous monologues," in *Proc. of IEEE Workshop on Speech Synthesis*, 2002, pp. 203–206.

[3] S. King and V. Karaiskos, "The Blizzard Challenge 2012," in *Proc. of Blizzard Challenge 2012 Workshop*, 2012.

[4] G. Bailly and C. Gouvernayre, "Pauses and respiratory markers of the structure of book reading," in *Proc. of Interspeech*, 2012.

[5] N. Braunschweiler, M. Gales, and S. Buchholz, "Lightly supervised recognition for automatic alignment of large coherent speech recordings," in *Proc. of Interspeech*, 2010, pp. 2222–2225.

[6] S. Buchholz, J. Latorre, and K. Yanagisawa, "Crowd sourced assessment of speech synthesis," in *Crowd Sourcing for Speech Processing Applications to Data Collection, Transcription and Assessment*. Wiley & Sons, 2012.

[7] ESPS/waves+, "Manuals of product release 5.3," in *Entropic Inc., Washington, DC*, 2001.

[8] P. Taylor, *Text-to-Speech Synthesis*. Cambridge University Press, 2009.

Role of Pausing in Text-to-Speech Synthesis for Simultaneous Interpretation

Vivek Kumar Rangarajan Sridhar, John Chen, Srinivas Bangalore, Alistair Conkie

AT&T Labs - Research
180 Park Avenue, Florham Park, NJ 07932, USA

Abstract

The goal of simultaneous speech-to-speech (S2S) translation is to translate source language speech into target language with low latency. While conventional speech-to-speech (S2S) translation systems typically ignore the source language acoustic-prosodic information such as pausing, exploiting such information for simultaneous S2S translation can potentially aid in the chunking of source text into short phrases that can be subsequently translated incrementally with low latency. Such an approach is often used by human interpreters in simultaneous interpretation. In this work we investigate the phenomena of pausing in simultaneous interpretation and study the impact of utilizing such information for target language text-to-speech synthesis in a simultaneous S2S system. On one hand, we superimpose the source language pause information obtained through forced alignment (or decoding) in an isomorphic manner on the target side while on the other hand, we use a classifier to predict the pause information for the target text by exploiting features from the target language, source language or both. We contrast our approach with the baseline that does not use any pauses. We perform our investigation on a simultaneous interpretation corpus of Parliamentary speeches and present subjective evaluation results based on the quality of synthesized target speech.
Index Terms: simultaneous interpretation, translation, pausing, prosody, mean opinion score (MOS)

1. Introduction

Simultaneous interpretation (SI) refers to the challenging task of listening to speech in the source language and simultaneously interpreting (non-verbatim translation) it in the target language. Even though simultaneous interpreters have been providing satisfactory services daily in dozens of languages and thousands of meetings across the world (e.g., United Nations, embassies, etc.), it is an arcane art that has received little attention from the speech and language research community. One of the critical constraints in SI is that the delay between a source language chunk and its corresponding target language chunk (referred to as *ear-voice-span*) is kept minimal in order to continually engage the listeners. Simultaneous interpreters are able to generate target speech incrementally with very low ear-voice span by using a variety of strategies [1] such as anticipation, cognitive and linguistic inference, paraphrasing, etc. As a consequence, the translated segments can range from short phrases to a complete sentence.

Simultaneous translation using speech translation technology has been gradually trying to reduce the dependence on human interpreters to improve the scalability as well as eliminate the fatigue associated with prolonged human interpretation. However, target language synthesis in such systems is either ignored; i.e., only speech-to-text is enabled, or performed at the sentence level using the translated text. The notion of an utterance is typically obtained by predicting punctuation on the source text, translating the sentence and subsequently synthesizing the complete sentence using text-to-speech synthesis. Such an approach loses the rich information contained in the source speech signal that may be vital for incremental translation. Simultaneous Interpreters use several acoustic and prosodic cues from the source speech to perform linguistic inference as well as control the pace of speech production in the target language [1]; e.g., taking a breath or perform planning during a source language pause, pausing in the target language to wait for the verb in the source language, etc. Disregarding such information, especially in speech-to-speech (S2S) translation of long speeches (talks and lectures), may result in monotonous speech synthesis of long segments that may impair the understanding of target speech.

In this work we investigate the phenomena of pausing in simultaneous interpretation and examine the impact of utilizing such information for target language text-to-speech synthesis in a simultaneous S2S system. We contrast different strategies for incorporating pause information in the target language. On one hand, we superimpose the source language pause information obtained through forced alignment (or decoding) in an isomorphic manner on the target side while on the other hand, we use a classifier to predict the pause information for the target text by exploiting features from the target language, source language or both. We perform our investigation on a simultaneous interpretation corpus of Parliamentary speeches and present subjective evaluation results based on the quality of synthesized target speech.

The rest of the paper is organized as follows. In Section 2 we formally define the problem and describe the data used in this work in Section 3. We describe the experimental setup in Section 4 followed by results of the experiments in Section 4.3. We provide a brief discussion about the experimental results in Section 5 followed by conclusions and directions for future work in Section 6.

2. Problem Formulation

The basic problem of text translation can be formulated as follows. Given a source (French) sentence $\mathbf{f} = f_1^J = f_1, \cdots, f_J$, we aim to translate it into target (English) sentence $\hat{\mathbf{e}} = \hat{e}_1^I = \hat{e}_1, \cdots, \hat{e}_I$.

$$\hat{\mathbf{e}}(\mathbf{f}) = \arg\max_{\mathbf{e}} \Pr(\mathbf{e}|\mathbf{f}) \quad (1)$$

If, as in talks, the source text (reference or ASR hypothesis) is very long, i.e., J is large, we attempt to break down the source string into shorter sequences, $\mathbf{S} = s_1 \cdots s_k \cdots s_{Q_s}$, where each sequence $s_k = [f_{j_k} f_{j_k+1} \cdots f_{j_{(k+1)}-1}]$, $j_1 = 1, j_{Q_s+1} = J+1$. Let the translation (or interpretation) of each foreign sequence s_k be denoted by $t_k = [e_{i_k} e_{i_k+1} \cdots e_{i_{(k+1)}-1}]$, $i_1 =$

$1, i_{Q_s+1} = I^{'} + 1$[1]. The segmented sequences can be translated using a variety of techniques [2] while the segmentation itself can be obtained using linguistic and non-linguistic strategies [3, 4, 5]. The translated sequence, $\mathbf{T} = t_1 \cdots t_k \cdots t_{Q_s}$, is typically synthesized independently using a text-to-speech synthesizer that generates appropriate prosody and pausing using pre-trained models.

Our objective is to improve the quality of speech synthesis in the above framework by predicting pausing information for the translated sequence $\mathbf{T}$; i.e., for the output sequence $t_1 \cdots t_{Q_s} = [e_1 \cdots e_{I'+1}]$, we predict the presence or absence of silence (binned into N intervals) between each pair of words. Subsequently, the new silence inserted sequence $[e_1\ sil_1\ e_2\ sil_2\ e_3\ nosil_3 \cdots sil_{I'}\ e_{I'+1}]$ is used by the TTS engine; $sil_1, sil_2, nosil_3, sil_{I'}$ are the predicted classes in the example. Since we can get the word alignment information of a partially translated sequence, it is feasible to bootstrap source language silence information (obtained from a speech recognizer) as well as other possible syntactic information associated at a word level in the target language prediction. In training a classifier to predict pauses for the target language, one can use a variety of target as well as source language features, thus, facilitating inference from the source language signal.

We use a maximum entropy classifier for predicting the silence class after each target word. Given a sequence of translated words $e_1 \cdots e_{I'+1}$, their parts of speech (POS) $p_1 \cdots p_{I'+1}$, their corresponding source words $f_1 \cdots f_{J+1}$, and a pause label vocabulary ($l_i\ \epsilon\ \mathcal{L}, |\mathcal{L}| = N + 1$), the best pause label sequence $L^* = l_1, l_2, \cdots, l_{I'}$ is obtained by approximating the sequence classification problem, using conditional independence assumptions, to a product of local classification problems as shown in Eq.(3). The classifier is then used to assign a pause label to each target word conditioned on a vector of local contextual features from both source and target sides.

$$L^* = \arg\max_L P(L|e_1 \cdots e_{I'+1}, p_1 \cdots p_{I'+1}, f_1 \cdots f_{J+1}) \tag{2}$$

$$\approx \arg\max_L \prod_{i=1}^{n} p(l_i|e_1 \cdots e_{I'+1}, p_1 \cdots p_{I'+1}, f_1 \cdots f_{J+1}) \tag{3}$$

$$= \arg\max_L \prod_{i=1}^{n} p\left(l_i|\mathbf{\Phi}_i(e_1 \cdots e_{I'+1}, p_1 \cdots p_{I'+1}, f_1 \cdots f_{J+1})\right) \tag{4}$$

where $\mathbf{\Phi}_i(e_1 \cdots e_{I'+1}, p_1 \cdots p_{I'+1}, f_1 \cdots f_{J+1})$ is a set of features extracted within a bounded local context around word e_i.

In order to obtain POS tags for words $e_1 \cdots e_{I'+1}$, a unigram POS tagger was implemented which used word shape features to predict the POS of unknown words. The English tagger was trained on the Penn Treebank while the Spanish tagger was trained on EPIC corpus (Section 3) tagged using Spanish Freeling [6].

3. Data

In order to train the target language pause classifier, one needs a corpus that contains source speech and its corresponding target speech (either translation or interpretation). We used the European Parliamentary interpretation corpus (EPIC). The EPIC corpus [7] is a parallel corpus of European Parliamentary speeches and their corresponding simultaneous interpretations. The source speeches are either in English (81), Spanish (21) or Italian (17) and each source speech is simultaneously interpreted in two other languages. We extracted the audio from the video clips of each source language speaker while the audio for the interpreted target speeches was already provided. The corpus also contains the transcripts of all the speeches. We use only the English-Spanish portion of the corpus; i.e., the 81 speeches interpreting from English to Spanish and 21 speeches with interpretation from Spanish to English. The genre of the speeches is also provided with the corpus and can be read, impromptu or spontaneous. We picked one speech from each of these categories for testing and used the remaining for training.

As a first step in our analysis we forced aligned the English and Spanish speeches independently using generic acoustic models. The English acoustic model was trained on about 600 hours of TED talks while the Spanish acoustic model was trained on close to 1000 hours of speech collected through smartphones. Both the acoustic models were trained using minimum phone error (MPE) criterion using the AT&T WATSON^{SM} speech recognizer [8]. The resulting word segmentation contained the start and end duration for each word as well as silences (with duration). Subsequently, we aligned the transcripts in the parallel speeches at the sentence level using dynamic programming with an English-Spanish dictionary.

3.1. Inducing word alignment

Unlike parallel text used in building word and phrase-based machine translation models, SI texts maybe non-parallel and even non-comparable. As a result, inducing word correspondence using automatic word alignment is quite difficult. First, we used a sentence matching algorithm [9] to align the sentences across the two languages. Subsequently, we used a custom algorithm for aligning the words across the two languages. The matching was facilitated by a dictionary obtained through automatic alignment [10] of a large English-Spanish parallel corpus comprising of about 8 million sentence pairs. The resulting dictionary was filtered such that only top 10 target translations (sorted by posterior probability) of each source word was preserved in the final dictionary.

Our word alignment procedure links each source word with its closest matching target word, if possible, according to heuristics. These heuristics take into account the amount of time between when the source word is spoken and its corresponding target word is spoken, as well as translation probabilities obtained through the dictionary. Specifically, the input consists of a sequence of source words $(f_1, f_2, \ldots, f_J)$ and a corresponding sequence of target words $(e_1, e_2, \ldots, e_I)$. In addition, there is a function TIME that maps a source or target word to its start time and another function STOP that maps a source or target word to *true* if it is a stopword and *false* otherwise. Finally, it is assumed that translation probabilities $P(e_i|f_j)$ are available.

The procedure takes three parameters: δl and δr define the left and right part of the time window in which the target word e_i corresponding to the source word f_j is taken to appear. t is a probability threshold that forbids a target word e_i from linking to a source word f_j when $P(e_i|f_j) < t$. For our experiments, we chose $\delta l = 1$ second, $\delta r = 6$ seconds, and $t = 0.008$. The procedure tries to link each source word $f_j \in (f_1, \ldots, f_J)$ to a target word as follows. First, a candidate set F_e of target words is constructed such that $e_i \in (e_1, \ldots, e_I)$ is placed in F_f if and

[1]The segmented and unsegmented talk may not be equal in length, i.e., $I \neq I^{'}$

only if the following criteria hold:

- TIME(f_j) $-$ $\delta l \leq$ TIME(e_i) $\leq$ TIME(f_j) $+$ δr
- STOP(f_j)$\wedge$ STOP(e_i) or $\neg$STOP(f_j) $\wedge$ $\neg$STOP(e_i)
- $P(e_i|f_j) \geq t$

Finally, e_i^* is output where,

$$e_i^* = \arg\max_{e_i \in F_f} P(e_i|f_j) \quad (5)$$

4. Experimental Setup

We examine the utility of predicting pauses in target language for improved text-to-speech synthesis using five different stimuli. The stimuli used in our investigation is as follows.

s1: Target text separated by reference punctuation (only period)

s2: Target text with pauses obtained through forced alignment of reference target text

s3: Target text with pauses superimposed from forced alignment of reference source text

s4: Target text with pauses predicted using a classifier trained on target language features

s5: Target text with pauses predicted using a classifier trained on source and target language features

In the first stimulus **s1**, manual transcription of the interpreted speech marked with sentence boundaries is used for synthesis. We only use periods as markers of sentence boundary. In simultaneous speech-to-speech translation systems, one typically gets such an output albeit with errors introduced during automatic speech translation. In the second stimulus **s2**, we take the forced alignment of the target text obtained by using a speech recognizer and insert pauses into the text as determined by the ASR; i.e., the pausing is identical to that used by the interpreter during the target speech production. The stimulus **s3** is an isomorphic mapping of pauses from the source to the target. We project the silences obtained through forced alignment of the source speech onto the target through the word alignment procedure described in Section 3.1. Since, the interpretation procedure does not generate a perfectly parallel text, some of the words in source and target may be unaligned. We superimpose the silences only on words that are aligned using our alignment procedure.

The stimuli **s4** and **s5** are created by inserting pauses predicted automatically through a classifier. Classifiers for both **s4** and **s5** predict pauses using the following pause label vocabulary:

Label	**Meaning**
no silence	$0 \leq$ pause $<$ 0.2 sec
short break	0.2 sec $\leq$ pause $<$ 0.5 sec
long break	0.5 $\leq$ pause

Table 1: Description of the classes used in the classifier

Pauses in the EPIC corpus were mapped to these pause labels as follows: pauses less than 0.2 seconds were mapped to *no silence*; pauses between 0.2 and 0.5 seconds were mapped to *short break*; and pauses greater than 0.5 seconds were mapped to *long break*.

Feature sets $\mathbf{\Phi}_i$ for classifiers for both **s4** and **s5** contain words and POS in a five word window around the target word e_i to be tagged. In addition, feature set $\mathbf{\Phi}_i$ for the classifier for **s5** contained two features encoding the types of pauses, if any, that occurred before and after source word f_i to which the target word e_i has been linked.

Classifiers for **s4** and **s5** were trained on 18 speeches (source: Spanish) from the EPIC corpus and tested on 3 other speeches of this type. Results are shown below in Table reftable:classification.

	Class	*Recall*	*Precision*	*F*
s4	*no silence*	0.9811	0.8630	0.9182
	short break	0.0374	0.2667	0.0656
	long break	0.1452	0.3600	0.2069
s5	*no silence*	0.9821	0.8631	0.9188
	short break	0.1294	0.3333	0.0960
	long break	0.1452	0.4286	0.2169

Table 2: Classification performance of the classifiers used for generating stimuli **s4** and **s5**

Overall, the classification results indicate that it is quite difficult to predict short and long breaks in comparison with absence of silence. Classifier **s5** performs somewhat better than **s4**, showing that silence information from the source speech helps predict silence in the target. **s5** encoded only a small amount of such information as features; adding more information from the source speech may improve the classifier's accuracy further. In addition, the results may be skewed because the training data for our classifier is quite sparse. There were only 18 speeches interpreted from Spanish-English. As part of our current study, we are performing experiments for English-Spanish that has larger amounts of training data but require Spanish speakers to take the listening tests.

4.1. Experimental Design

The Web-based listening tests were administered in two ways: Web interface hosted on a standalone server and Amazon Mechanical Turk. We picked three speeches from the EPIC corpus; Spanish source speech interpreted into English as we had access to more English speakers for subjective listening tests. The three speeches belonged to read, impromptu and mixed genre categories to cover varying styles of the speeches. Since the source speeches were 1.5 minutes long, it was deemed that using the entire speech was too cumbersome for a listener to listen to during a listening test. Hence, we selected two 30 second snippets from each speech. The final listening test was comprised of 6 audio snippets across the five stimuli.

The listening test had 6 sections with each section comprising 5 stimuli. The listeners were asked to rate each audio file on a scale of 1-5 (bad, poor, fair, good, excellent). The listeners also indicated whether or not English was their native language, and whether they listened using headphones or speakers.

4.2. Listeners

A total of 100 listeners participated in the subjective listening test; 74 were native English speakers while 26 were non-native English speakers. Furthermore, 88 listeners took the test using headphones and the remaining 12 used their PC speakers. The average time taken for the test was 19 minutes (the minimum time to listen to all the stimuli is 30*0.5minutes=15 minutes).

4.3. Experimental Results

The results of the subjective listening test is summarized in Table 3. The table shows the mean and standard deviation of the ratings overall as well as across the 3 genres of speech (read, mixed and impromptu). The results indicate that the listeners prefer the synthesized audio from reference punctuation for the target text. However, the average length of a sentence in the test set is 19 words which is prohibitively long for synthesis in simultaneous S2S interpretation or translation. The average length of a sentence for **s2**, **s3**, **s4**, **s5** is 3, 4, 8 and 7 words, respectively. The quality of synthesis for long sentences is presumably better as the TTS engine can use longer units as well better prosody. The quality of synthesis for the other stimuli is mostly fair but significantly poorer than stimulus **s1**. It is also interesting that the quality for impromptu speech is better than that for the read and mixed mode of speech. When the speech is unplanned and more informal, the pauses predicted by the classifier are acceptable to the listener in contrast with read speech that has typically has a rigs syntactic structure. The results in general indicate that pauses either superimposed from the source speech or predicted using a classifier (target or source and target features) can offer a reasonable means of synthesizing target speech incrementally in a S2S translation setting. Considering that stimulus **s1** cannot be used in a real-time translation scenario, we need to balance latency versus synthesis quality using the approaches presented through stimuli **s2-s5**.

Stimulus	Rating (mean and standard deviation)			
	Overall	Read	Mixed	Impromptu
s1	3.6±0.9	3.5±0.9	3.7±0.8	3.6±0.9
s2	2.9±1.0	2.7±1.1	2.8±1.0	3.1±0.9
s3	2.9±1.0	2.7±1.0	3.0±1.0	3.1±0.9
s4	2.8±1.0	2.8±1.0	2.7±1.0	3.0±0.9
s5	2.9±1.0	2.9±1.0	2.9±0.9	3.0±1.0

Table 3: Mean and standard deviation of ratings across the five stimuli

5. Discussion

The experiments performed in this work are on reference text; i.e., no translation system was used for translating the source text into target language. Hence, it is the ideal case scenario where one can assume perfect translation (or interpretation). The accuracy of the pause classifier is bound to degrade while operating with noisy text translations. We plan to perform this investigation as part of future work.

The pause classifier predicts non-pauses reasonably well, but predicts pauses with poor accuracy. Part of the reason is the small amount of training data (18 speeches, about 41,500 words). Also, within this data there are about 9,500 examples of non-pauses and only 1,500 examples of pauses, which may explain the impoverished accuracy of pause prediction. Boosting methods which may better delineate the decision boundary for pauses may bring up their accuracy. In the case of **s4** since the prediction is based only on target text, one can conceivably use a large amount of non interpretation data to learn the model. However, the model is likely to predict pauses as in prepared speeches in contrast with simultaneously interpreted target speech.

Another problem with the prediction of pauses is that instead of having several local maxima in the distribution of sorted pauses in the training data to which one might assign discrete pause labels such as *short pause* or *long pause*, the distribution is a smooth curve that exponentially decreases as pause time increases. Thus, the binning of pauses into discrete labels that were done for these experiments were somewhat arbitrary.

The training data used to train the pause classifier is limited in this work as we had only 18 speeches from Spanish-English. We are currently performing experiments for Ensligh-Spanish with larger amount of training data (81 speeches). It can be expected that the classifier accuracy will increase with larger amounts of training data.

6. Conclusion

In this work we investigated the phenomena of pausing in simultaneous speech interpretation and studied the problem of using such information for target language text-to-speech synthesis in a simultaneous speech-to-speech translation system. We contrasted several ways of predicting pauses in the target language for a speech-to-speech translation setting, particularly, speech interpretation from Spanish-English. Our results indicate that either superimposing source language pauses or predicting pauses for the target language by exploiting lexical and syntactic features (both source and target language) can result in reasonably good quality synthesized speech when the input speech is unplanned, i.e., impromptu. However, the quality of synthesis suffers when the input speech is read as the speaker pauses less often. Our results also indicate that pauses can be used as good markers for chunking the source speech to reduce the latency in speech-to-speech translation. We are currently performing experiments on a larger corpus as well as analysis in English-Spanish (resulting in Spanish text-to-speech synthesis).

7. References

[1] G. V. Chernov, *Inference and anticipation in simultaneous interpreting*. John Benjamins, 2004.

[2] S. Bangalore, V. K. Rangarajan Sridhar, P. Kolan, L. Golipour, and A. Jimenez, "Real-time incremental speech-to-speech translation of dialogs," in *Proceedings of NAACL:HLT*, June 2012.

[3] M. Cettolo and M. Federico, "Text segmentation criteria for statistical machine translation," in *Proceedings of the 5th international conference on Advances in Natural Language Processing*, 2006.

[4] C. Fügen, A. Waibel, and M. Kolss, "Simultaneous translation of lectures and speeches," *Machine Translation*, vol. 21, pp. 209–252, 2007.

[5] C. Fügen and M. Kolss, "The influence of utterance chunking on machine translation performance," in *Proceedings of Interspeech*, 2007.

[6] L. Padr and E. Stanilovsky, "Freeling 3.0: Towards wider multilinguality," in *Proceedings of the Language Resources and Evaluation Conference (LREC 2012)*. Istanbul, Turkey: ELRA, May 2012.

[7] C. Bendazzoli and A. Sandrelli, "An approach to corpus-based interpreting studies," in *Proceedings of the Marie Curie Euroconferences MuTra: Challenges of Multidimensional Translation*, Saarbrücken, 2005.

[8] V. Goffin, C. Allauzen, E. Bocchieri, D. Hakkani-Tür, A. Ljolje, and S. Parthasarathy, "The AT&T Watson Speech Recognizer," Tech. Rep., September 2004.

[9] V. K. Rangarajan Sridhar, L. Barbosa, and S. Bangalore, "A scalable approach to building a parallel corpus from the Web," in *Proceedings of Interspeech*, 2011.

[10] F. J. Och and H. Ney, "A systematic comparison of various statistical alignment models," *Computational Linguistics*, vol. 29, no. 1, pp. 19–51, 2003.

Minimum Error Rate Training for Phrasing in Speech Synthesis

Alok Parlikar and Alan W Black

Language Technologies Institute, Carnegie Mellon University, Pittsburgh, PA. USA
{aup, awb} @cs.cmu.edu

Abstract

Phrase break prediction models in speech synthesis are classifiers that predict whether or not each word boundary is a prosodic break. These classifiers are generally trained to optimize the likelihood of prediction, and their performance is evaluated in terms of classification accuracy. We propose a minimum error rate training method for phrase break prediction. We combine multiple phrasing models into a log-linear framework and optimize the system directly to the quality of break prediction, as measured by the F-measure. We show that this method significantly improves our phrasing models. We also show how this framework allows us to design a knob that can be tweaked to increase or decrease the number of phrase breaks at synthesis time.

Index Terms: Speech Synthesis, Phrasing

1. Introduction

Phrase break prediction (phrasing) is an important prosodic step during speech synthesis. Other prosody models depend on phrasing decisions, and hence appropriate phrase breaks are critical to overall naturalness of synthetic speech. The problem of phrasing can be thought of as a classification problem: given some text, we want to classify each word boundary as being a phrase break or not. In terms of the TOBI[1] systems for prosody annotation, phrasing classifiers are typically trained to predict levels 1, 3, and 4. Phrasing classifiers are often trained using standard corpora: e.g., the Festival[2] system uses a model[3] trained on the MARSEC[4] data for English voices. If manually annotated data is not available, phrasing models can be trained by force-aligning the speech and text data available for building synthetic voices[5].

In practice, phrasing models can be decomposed into two disparate models: (i) A model that takes local context of a word boundary into account to decide how likely a break is, and (ii) A model that takes longer context of previously generated breaks to decide how frequently breaks should be generated. These two models can be combined together, such as using the Viterbi method, to decide the optimal sequence of phrase breaks.

This paper discusses two important aspects of phrasing, and attempts to build upon the state of the art: (i) Optimization target for phrasing, and (ii) Phrasing and changes in speaking rate.

Phrasing classifiers are typically trained to maximize the likelihood of break prediction. However, they are then evaluated on the basis of an accuracy measure, such as F-1 score[6]. An objective improvement in F-1 score is also perceived by people in subjective listening tests. In order to make a higher perceptual impact, we aim to optimize our phrasing model directly to the F-1 score.

Phrasing models are usually insensitive to the speaking rate. In natural speech, we observe that fast speech has fewer phrase breaks, and slower speech tends to have more breaks. A phrasing model needs to thus provide a mechanism that allows us to insert more or fewer breaks at synthesis time depending on the speaking rate.

We propose a minimum error rate training approach that provides a solution to both these needs in phrasing. The idea here is to combine multiple phrasing models in a log-linear fashion, and learn weights for different models in order to maximize the F-measure of break prediction. We describe how such a framework not only improves the break prediction, but also offers a "knob" to increase or decrease the number of predicted breaks. We also discuss how this framework can be useful in solving other difficult problems, such as phrasing in the context of the synthesis of disfluent machine translation output.

2. Classic Phrasing (Baseline)

Given a text corpus annotated with breaks, there are several ways of training a phrasing model. Rule based methods[7], or data driven methods using machine learning techniques such as decision trees[8, 9], transformational rule learning[10], z-score models[11], hidden Markov models[3, 12, 13], memory based learning[14], Baysian networks[15], maximum entropy models[16], and neural networks[17] have been successful at the task. In this work, we use the models[3, 5] that have been setup for the Festival speech synthesis system[2].

Festival's phrasing uses two models: a context model, and a sequence model. If b_i is the probability of a break at the juncture i, C_i is the context of features at juncture i, and B_i represents the context of previous break sequences at juncture i, then we want to estimate $P(b_i|C_i, B_i)$. With the help of the Bayes theorem, and a few independence assumptions, we can derive that

$$P(b_i|C_i, B_i) \propto \frac{P(b_i|B_i) \cdot P(b_i|C_i)}{P(b_i)}.$$

The term $P(b_i|B_i)$ is essentially the language model probability of a given break sequence. We estimate this in Festival using a 7-gram model[3]. The term $P(b_i)$ is the unigram probability of a word boundary being a break. Our context model, $P(b_i|C_i)$, is a grammar based model[5]. This is estimated using a decision tree classifier, that uses word level features, positional features, and syntactic features. Given these models, at synthesis time, our goal is to find the optimal break sequence b^*:

$$b^* = \arg\max_b \prod_i P(b_i|C_i, B_i)$$

$$\therefore b^* = \arg\max_b \prod_i P(b_i|C_i) \cdot P(b_i|B_i)$$

.

In order to find the most likely break sequence, Festival runs a Viterbi search over the possible phrase break sequences. This is schematically shown in Figure 1.

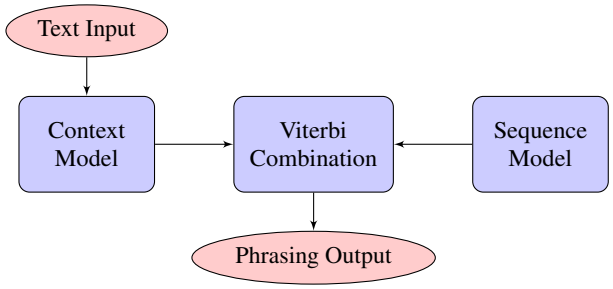


Figure 1: Viterbi Phrasing Strategy in Festival

3. Minimum Error Rate Training

Ideally we want to train our phrasing model such that the end-to-end performance in perceived synthesis quality is optimal. A model trained to maximize the likelihood of phrase breaks makes the simplifying assumption that the final evaluation is based on simply counting the number of wrong decisions made. However, a metric such as F-measure is a little more complex, since it takes both precision and recall into account, and is better correlated to perception than just the accuracy of phrasing. Our goal is to optimize the selection of phrase breaks in order to minimize the error our model makes, as measured by the F-1 score. The idea of using minimum error-rate training (MERT) in phrasing is inspired from its use in the Statistical Machine Translation[18].

Let us assume that we are given a text sequence $\mathbf{t}$, and we want to produce a break sequence $\mathbf{b}$. Among all possible break sequences, we will choose the sequence with the highest probability:

$$\mathbf{b}^* = \arg\max_b P(\mathbf{b}|\mathbf{t}).$$

We directly model the posterior probability $P(\mathbf{b}|\mathbf{t})$ using a log-linear model. In this framework, we have a set of M feature functions, $h_m(\mathbf{b}, \mathbf{t})$. For each feature function, we have a weight w_m. The direct phrasing probability is then given by:

$$P(\mathbf{b}|\mathbf{t}) = \frac{\exp\left(\sum_{m=1}^{M} w_m h_m(\mathbf{b}, \mathbf{t})\right)}{\sum_{b'} \exp\left(\sum_{m=1}^{M} w_m h_m(\mathbf{b}', \mathbf{t})\right)}.$$

The modeling problem here is to define suitable feature functions that capture the relevant properties of the phrasing task. The training problem is to find out suitable weights w_1^M. However, as mentioned before, we want to train the model to minimize error.

We define a held out development corpus D_1^N, of size N, with text sequences T_1^N that has reference break annotations R_1^N. Our goal is to obtain minimum error on this corpus, and a set of k different candidate break sequences, $S_n = \mathbf{b}_{n,1}, \ldots, \mathbf{b}_{n,k}$. That is, for each of the N sentences in the test set, we have k hypotheses of break sequences, and we want to pick the ones that minimize overall error on the test set. Given a set of weights w_1^M, the top-best break sequence $\mathbf{b}_n$ for sentence n is given by:

$$\mathbf{b}_n = \arg\max_{\mathbf{b} \in S_n} \left[\sum_{m=1}^{M} w_m h_m(\mathbf{b}|T_n) \right].$$

For each sentence in the development corpus D_1^N, we can pick the best break sequence given some weights, and then compute the F-measure over these break sequences. The error function E can then be set to negative value of the F-measure. If $E(D_1^N; w_1^M)$ represents the error on the test set given a set of weights, we have:

$$w_1^M * = \arg\min_{w_1^M} \left[E(D_1^N; w_1^M) \right].$$

The optimization criterion here is tricky. Because of the presence of an $\arg\max$ operation within the Error function, we can not compute the gradient of the error, and hence an optimization method such as gradient descent can not be used here. The error surface is not smooth, and has many local minima.

We use the Basin-Hopping algorithm[19] to optimize the error function at hand. This global minimization method has been shown to be extremely efficient for a wide variety of problems, and is especially useful when the error function has many minima separated by large barriers. In particular, we use the implementation of this algorithm within the Python SciPy toolkit.

We use a development corpus, use a randomly initialized weight sequence and produce an n-best list of break sequences. We then run the minimum error rate training over these n-best sequences and learn new weights. We then use the new weights and re-generate an n-best list of break sequences over the development corpus, and run minimum error rate training again. We repeat these iterative process until the final error does not improve across an iteration. After each iteration, we also normalize the weight vector to be of a unit norm.

We use four feature functions $h_m(\mathbf{b}|\mathbf{t})$ in our method. Two of these are the same as the models in the baseline phrasing method: (i) The context model $P(b_i|C_i)$ that looks at the lexical and syntactic context, and (ii) $P(b_i|B_i)$ that looks at the language model probability of the break sequence. In addition, we use another context model, $P(b_i|C_i)$, defined in [3] that looks at the part-of-speech tag context at a word boundary and uses a quadgram Language Model to predict the probability of the word boundary being a break. Finally, we use a break-count feature, that counts the total number of breaks in the predicted break sequence.

4. Experimental Results

We evaluated our method on two synthetic voices trained using the CLUSTERGEN[20] statistical parametric synthesis method: (i) Voice built from about an hour of speech in the F2B corpus within the Boston University Radio News Corpus[21], and (ii) Voice built from two hours of recordings of Jane Austen's books, for Blizzard Challenge 2013 task EH2. We split the corpora into splits of 80-10-10 for training, development and testing.

Our baseline phrasing models were built to be style-specific phrasing models[5] in each case. We trained the proposed model with minimum error rate method on a held out corpus, and used the unseen test partition in the same domain to compare the baseline method to the proposed approach. Table 1 shows the comparison of the models in terms of the F-1 metric[6]. We see that the proposed method yields an improvement over the baseline on both datasets.

5. Phrasing Rate: "Knob"

One requirement of a phrasing model is that it should be flexible to adapt to the speaking rate of a synthesizer. A slow synthesizer should probably mark more word boundaries as breaks, and a faster synthesizer can do away with a few breaks. If the user of a speech synthesis engine demands that 30%, or 60% of the word

Table 1: Objective Evaluation (F-1 measure) of the Proposed MERT Method. Improvements are significant at $p < 0.05$

Voice	Baseline	Proposed
F2B	54.35	58.06
Audiobook	52.87	57.58

boundaries should be breaks, then our phrasing model should be able to meet this requirement. However, this is a tricky constraint. If our training data had splits corresponding to slower and faster speaking styles, we could train individual classifiers and use the appropriate one at synthesis time. But such data is seldom available, and collecting data to train such specific models is difficult. We describe how we use the log-linear framework and MERT mechanism to provide a knob, a continuous number, to vary the number of phrase breaks produced.

One of the features that we used in the log-linear model was simply the number of phrase breaks in a given break sequence. This feature allows us to define a knob to change the number of phrase breaks our model produces.

Intuitively, the break-count feature tries to make sure that the number of breaks produced by our model is reasonably close to the number of breaks in the reference sequences in our development data. Even if we optimize towards the F-measure of break prediction, which itself balances precision and recall of phrasing, having this additional feature means that the weight learned for this feature will produce an optimal number of phrase breaks. If we keep the weights for other features to be the same, and change the weight of the break-count feature, then the search process at synthesis time picks utterances with more or fewer breaks than the optimal. For example, if we subtract a number from the weight of the break count feature, and maximize the log-linear combination, we would produce more breaks. We can vary the value of this weight and measure the number of word boundaries in a development corpus that were breaks. The weight of the break-count feature is thus the knob we can use to tweak the amount of phrasing. Figure 2 shows this curve for the two voices we have. The x-axis shows the value of the knob (i.e., the weight of the break-count feature) and the y-axis shows what percentage of word boundaries in a corpus were predicted as being breaks.

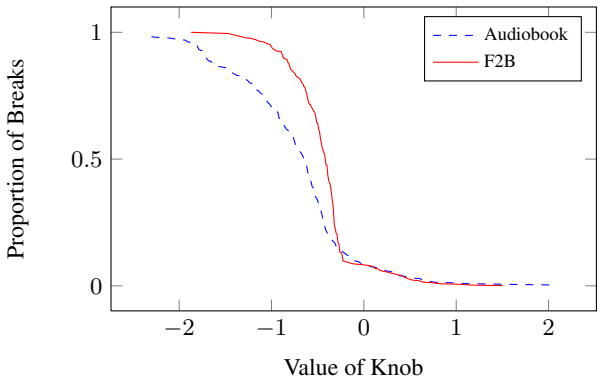


Figure 2: Proportion of phrase breaks generated by varying the log-linear weight of the break-count feature (the knob)

In order to customize the phrasing rate at demand, we need to parameterize the "knob", so that given a particular value of expected proportion of breaks, we can set an appropriate weight for the break-count feature during synthesis. This problem boils down to deriving an equation for the inverse of the function represented in Figure 2. Given a particular phrasing proportion x, we want to find out the value k that our knob should be set to.

To learn the parametric equation of the knob, we use a development corpus and varying values of the weight of the break-count feature to generate data points depicted in Figure 2. We then fit the data automatically to a variety of sigmoidal, trignometric and simple functions and choose the function that best fits the data we have, as measured by the root-mean-squared error of the fit. We used open-source fitting code, pyeq2[22] in this work.

Our empirical analysis shows that for various corpora and phrasing model combinations, the phrasing knob curve can be approximated, well within a RMS tolerance of about 0.15, using a Tangent equation with offset:

$$k = A \cdot \tan\left(\pi \frac{x - C}{W}\right) + O$$

where A (Amplitude), C (center), W (width) and O (offset) are the parameters we learn automatically. For the F2B voice, we obtained

$$k_{f2b} = -0.165 \cdot \tan\left(\pi \frac{x - 0.5073}{1.0772}\right) - 0.4670$$

and for the Audiobook voice, we obtained

$$k_{audiobook} = -0.2682 \cdot \tan\left(\pi \frac{x - 0.5048}{1.072}\right) - 0.8084$$

By tweaking the knob to change the phrasing rate, we deviate from the reference break sequences that we originally used to train our MERT model. This means, by changing the knob, we obtain fewer or more breaks, but at the cost of the F-measure. Of course, since the goal was to insert more or fewer breaks, the penalty in F-measure is not very relevant anymore, but we looked at what the drop in the F-measure looks like. Figure 3 shows how the F-measure changes when we set the expected break proportion to different values. We observe that the F-measure is highest when the knob is set to its original value, as learned from the MERT training.

6. Conclusions and Future Work

We described our method of defining the phrasing problem under a log-linear framework and training the framework with a minimum error rate target, rather than maximum likelihood. We showed that combining features/models related to phrasing using this MERT strategy produces a significant improvement in phrasing accuracy, as measured by the F-1 metric.

We described a break-count feature integral to our MERT model that allows us to define a parametric "knob" to vary the quantity of generated phrase breaks. Once we learn our MERT weights, we can keep all weights to their learned value and vary the weight of the break-count feature to provide this knob. Our empirical evidence shows that the knob can be reasonably approximated with a Tangent function with offset. The combination of using a MERT model and this break-count feature allows a user to specify how many breaks they want, and our model produces the breaks appropriately.

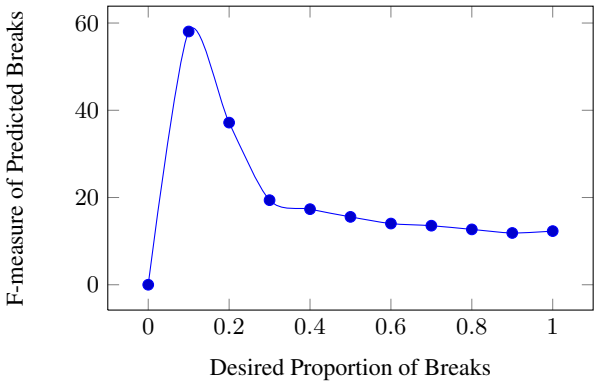


Figure 3: F-measure versus Desired Proportion of Phrase Breaks on the F2B corpus

We intend to explore the benefits of the proposed framework in more details. Particularly, we only used four feature functions in this work and would like to investigate how additional features could help improve the phrasing accuracy even further.

Our model now can vary the phrasing rate at demand. We map a particular phrasing proportion into a knob value. However, users of speech synthesis do not define phrasing rate in terms of the proportion of word boundaries that are breaks. The control we would like them to have would be more quantized: low, medium, high, etc. However, how does a particular category, such as "low" map into the desired proportion of break boundaries? We think this mapping depends on the original proportion of breaks in our reference break sequences. However, we intend to conduct listening tests to discover what grades of phrase breaks people can perceive, and how we can map categories of break levels into numeric proportion values.

The MERT framework that we proposed for phrasing took inspiration from work in Machine Translation. However, this connection actually runs deeper. Text to speech is often used as the final step in speech to speech translation, and we are required to synthesize automatically translated output. [23] has shown that the synthesis of machine translation output is often difficult to understand, and [24] suggests that appropriate phrasing can make it more understandable. We aim to use the MERT framework to incorporate internal machine translation scores of an utterance, that relate to confidence measures of the MT system, into the phrasing model and improve the intelligibility of synthesis.

7. References

[1] K. Silverman, M. Beckman, J. Pitrelli, M. Ostendorf, C. Wightman, P. Price, J. Pierrehumbert, and J. Hirschberg, "ToBI: A standard for labeling english prosody," in *Proceedings of 2nd International Conference on Spoken Language Processing*, Banff, Alberta, Canada, October 1992, pp. 867–870.

[2] A. W. Black and P. Taylor, "The festival speech synthesis system: system documentation," Human Communication Research Centre, University of Edinburgh, Tech. Rep., January 1997. [Online]. Available: http://www.cstr.ed.ac.uk/projects/festival

[3] P. Taylor and A. W. Black, "Assigning phrase breaks from part-of-speech sequences," *Computer Speech and Language*, vol. 12, pp. 99–117, 1998.

[4] P. Roach, G. Knowles, T. Varadi, and S. Arnfield, "MARSEC: A machine-readable spoken english corpus," *Journal of the International Phonetic Association*, vol. 23, no. 1, pp. 47–53, 1993.

[5] A. Parlikar and A. W. Black, "A grammar based approach to style specific phrase prediction," in *Proceedings of Interspeech*, Florence, Italy, August 2011, pp. 2149–2152.

[6] C. J. van Rijsbergen, *Information Retrieval*. Butterworth, 1979.

[7] J. Bachenko, E. Fitzpatrick, and C. Wright, "A computational grammar of discourse-neutral prosodic phrasing in english," *Computational Linguistics*, vol. 16, pp. 155–170, Sep. 1990. [Online]. Available: http://dl.acm.org/citation.cfm?id=98377.98380

[8] M. Q. Wang and J. Hirschberg, "Automatic classification of intonational phrase boundaries," *Computer Speech and Language*, vol. 6, pp. 175–196, 1992.

[9] P. Koehn, S. Abney, J. Hirschberg, and M. Collins, "Improving intonational phrasing with syntactic information," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, Istanbul, Turkey, June 2000.

[10] C. S. Fordyce and M. Ostendorf, "Prosody prediction for speech synthesis using transformational rule-based learning," in *Proceedings of the 5th International Conference on Spoken Language Processing*, Sydney, Australia, December 1998.

[11] P. A. Barbosa and G. Bailly, *Progress in speech synthesis*. New York: Springer Verlag, 1997, ch. Generation of pauses within the z-score model, pp. 365–381.

[12] H. Schmid and M. Atterer, "New statistical methods for phrase break prediction," in *Proceedings of the 20th international conference on Computational Linguistics*, Geneva, Switzerland, August 2004, pp. 659–665.

[13] N. Obin, P. Lanchantin, A. Lacheret, and X. Rodet, "Reformulating prosodic break model into segmental hmms and information fusion," in *Proceedings of Interspeech*, Florence, Italy, August 2011, pp. 1829–1832.

[14] E. Marsi, M. Reynaert, A. van den Bosch, W. Daelemans, and V. Hoste, "Learning to predict pitch accents and prosodic boundaries in dutch," in *Proceedings of Association for Computational Linguistics*, July 2003, pp. 489–496. [Online]. Available: http://dx.doi.org/10.3115/1075096.1075158

[15] M. Maragoudakis, P. Zervas, N. Fakotakis, and G. Kokkinakis, "A data-driven framework for intonational phrase break prediction," in *Text, Speech and Dialogue*, ser. Lecture Notes in Computer Science, V. Matoušek and P. Mautner, Eds. Springer Berlin / Heidelberg, 2003, vol. 2807, pp. 189–197.

[16] F. Liu, H. Jia, and J. Tao, "A maximum entropy based hierarchical model for automatic prosodic boundary labeling in mandarin," in *Proceedings of the 6th International Symposium on Chinese Spoken Language Processing*, Kunming, China, December 2008, pp. 1 –4.

[17] Z. Ying and X. Shi, "An RNN-based algorithm to detect prosodic phrase for chinese TTS," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, vol. 2, Salt Lake City, UT, USA, May 2001, pp. 809–812.

[18] F. J. Och, "Minimum error rate training in statistical machine translation," in *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics*, 2003, pp. 160–167.

[19] D. J. Wales and J. P. K. Doye, "Global optimization by basin-hopping and the lowest energy structures of lennard-jones clusters containing up to 110 atoms," *The Journal of Physical Chemistry A*, vol. 101, pp. 5111–5116, 1997.

[20] A. W. Black, "CLUSTERGEN: A statistical parametric synthesizer using trajectory modeling," in *Proceedings of Interspeech*, Pittsburgh, Pennsylvania, September 2006, pp. 194–197.

[21] M. Ostendorf, P. J. Price, and S. Shattuck-Hufnagel, "The boston university radio news corpus," Boston University, Tech. Rep., March 1995. [Online]. Available: http://ssli.ee.washington.edu/papers/radionews-tech.ps

[22] J. R. Phillips. Online curve and surface fitting. [Online]. Available: http://www.zunzun.com

[23] L. M. Tomokiyo, K. Peterson, A. W. Black, and K. A. Lenzo, "Intelligibility of machine translation output in speech synthesis," in *Proceedings of Interspeech*, Pittsburgh, September 2006.

[24] A. Parlikar, A. W. Black, and S. Vogel, "Improving speech synthesis of machine translation output," in *Proceedings of Interspeech*, Makuhari, Japan, September 2010, pp. 194–197.

HMM-based Speech Synthesis of Live Sports Commentaries: Integration of a Two-Layer Prosody Annotation

Benjamin Picart [1], *Sandrine Brognaux* [2], *Thomas Drugman* [1]

[1] TCTS - Université de Mons, Belgium
[2] Cental, ICTEAM - Université Catholique de Louvain, Belgium
benjamin.picart@umons.ac.be, sandrine.brognaux@uclouvain.be, thomas.drugman@umons.ac.be

Abstract

This paper proposes the integration of a two-layer prosody annotation specific to live sports commentaries into HMM-based speech synthesis. Local labels are assigned to all syllables and refer to accentual phenomena. Global labels categorize sequences of words into five distinct speaking styles, defined in terms of valence and arousal. Two stages of the synthesis process are analyzed. First, the integration of global labels (i.e. speaking styles) is carried out either using speaker-dependent training or adaptation methods. Secondly, a comprehensive study allows evaluating the effects achieved by each prosody annotation layer on the generated speech. The evaluation process is based on three subjective criteria: intelligibility, expressivity and segmental quality. Our experiments indicate that: (i) for the integration of global labels, adaptation techniques outperform speaking style-dependent models both in terms of intelligibility and segmental quality; (ii) the integration of local labels results in an enhanced expressivity, while it provides slightly higher intelligibility and segmental quality performance; (iii) combining the two levels of annotation (local and global) leads to the best results. It is indeed shown that it obtains better levels of expressivity and intelligibility.

Index Terms: HMM-based Speech Synthesis, Speaking Style Adaptation, Expressive Speech, Prosody, Sports Commentaries

1. Introduction

Expressive speech synthesis based on Hidden Markov Models (HMMs) has been the focus of many studies in the last ten years (e.g. [1], [2], [3] [4]). Conversely to unit-selection based synthesis, HMM-based synthesis [5] [6] offers a rich playground in terms of controllability of the generated speech. However, current research presents a certain number of drawbacks.

First, expressivity is often exclusively generated via adaptation or training on corpora with the targeted expressivity (e.g. [1], [2]). Generally, the training or the feature adaptation is achieved globally, with no consideration of local phenomena specific to expressive speech, like accentuation. However, it is widely acknowledged that the accentual structure of a sentence and the realization of focusses play a crucial role in the expressive function. Fonagy [7] notably emphasized the greater accentual density in emphatic speech. A few isolated studies (e.g. [8]) have tried to integrate some expressive accentual information in speech synthesis. However, they were led on Japanese, which is a language with more restricted accentual patterns compared to French or English [9].

Besides the omission of local phenomena, very few attention has been paid to macro prosodic changes. Indeed, most current studies rely on acted corpora of each emotion. These corpora are very constant regarding expressivity, being stereotypical with respect to the considered emotion. However, various expressivity types follow each other in real human speech. As stated by [10], "a coherent speech corpus includes prosodic effects that go beyond the sentence level". These global prosodic changes should be modeled to improve the quality of the generated expressive speech (see [11]).

The generation of an expressive prosodic realization is of utmost importance when synthesizing sports commentaries. Several studies have focused on their prosodic analysis (i.e. for basketball, football and rugby [12], horse races [13], soccer [14] and football [15] [16]). One of their main findings lies in the fact that such speech databases are importantly characterized by variations at the local but also at the global or macro level [15] [16]. At a macro level, [15] proposes to divide the corpus into three main speaking styles. *Elaboration* corresponds to relatively neutral speech. Conversely, dramatic style is more related to a high arousal level and can be subdivided into the *building up of a suspense*, which relates to a rise in the arousal level, and *the presentation of a highlight*, i.e. the arousal climax. These various speaking styles were shown to display specific prosodic features. It was pointed out in [12], for example, that highly excited phases, like shots, tend to be realized with a significantly higher fundamental frequency. This phenomenon was also observed in horse races at the end of the race, when the excitation reaches a maximum [13]. Interestingly, [16] emphasized the fact that, besides the arousal degree, the valence of the expressivity may also influence the prosodic realization of the commentaries. The analysis of sequences happening just after a goal in football games indicates, indeed, that the prosodic realization depends upon whether the goal is for or against the supported team [15]. This could be explained by the fact that sports commentaries are deeply 'listener-oriented' and that this acoustic distinction helps the listener decode the action more quickly. On the whole, most studies tend to suggest that a prosody annotation of sports commentaries requires, besides local accentual information, a more global annotation level assigning a specific speaking style to the speech segments.

This paper is in the continuity with our previous work on the subject [17], in which a prosody annotation protocol specific to sports commentaries (basketball in particular) and relying on two annotation levels was developed. A local annotation is associated to the syllable level and aims at annotating accentual events. A global annotation classifies groups of words into specific speaking styles. The interested reader is referred to [17] for more details. This annotation protocol was developed with HMM-based speech synthesis in view, which is implemented in this work.

One way to perform HMM-based speech synthesis is to

train a model, called *full data model*, using a database containing specific data (e.g. data corresponding to a particular emotion, degree of articulation of speech, etc.). Another way to build the models is to make use of adaptation techniques, which allow changing the voice characteristics and prosodic features of a source speaker into those of a target speaker [18]. These latter adapt the source HMM-based model with a limited amount of target speech data. The resulting model is called *adapted model*. The same concept holds for speaking style adaptation [19] [20]. This technique allows providing high quality speech synthesis using a limited amount of adaptation data [21].

Recently, Zen [22] proposed a new framework for estimating HMMs on data containing both multiple speakers and multiple languages. Speaker and language factorization attempts to factorize specific characteristics in the data and then models them using separate transforms. Another study [23] described a discrete/continuous HMM for modeling the symbolic and acoustic speech characteristics of speaking styles.

In this paper, we precisely aim at integrating efficiently the local and global annotations into an HMM-based speech synthesizer. The goal of this study is two-fold: i) quantifying the possible improvements brought by each annotation layer on various aspects of speech synthesis; ii) comparing different training methods regarding the integration of the global labels.

The paper is structured as follows. Section 2 presents the corpus used throughout this study. Section 3 summarizes the proposed annotation protocol and provides a brief overview of the acoustic analysis of both annotation levels (global and local). The integration of the proposed annotation protocol within HMM-based speech synthesis is investigated in Section 4 where some experiments are carried out in order to evaluate the quality of the generated speech across various aspects. Finally, Section 5 concludes the paper.

2. Database

This study is based on a corpus of live commentaries of two basketball games, uttered by a professional French commentator and recorded in sound-proof conditions. The speaker watched the game and commented it without any prompt. The speech signal was recorded with an AKG C3000B microphone. The audio acquisition system Motu 8pre was used outside the soundproof room, with a sampling rate of 44.1 kHz. The issue with sports commentaries corpora is usually the high level of background noise which precludes their precise acoustic analysis [15]. Conversely, our corpus exhibits the advantage of being spontaneous and of high acoustic quality, being therefore suited for speech synthesis. Both matches star the Spirou Belgian team with very tight final scores, which induces a high level of excitation. The corpus lasts 162 minutes, silences included.

The corpus was orthographically transcribed and the phonetization was automatically produced by [24], with manual check. The phonetic transcription was automatically aligned with the sound with [25], taking advantage of the bootstrap option to reach alignment rates higher than 80% with a 20 ms tolerance threshold. The Elite NLP system [26] produced other required annotation tiers (e.g. syllables, parts of speech, rhythmic groups, etc.). Sentence boundaries form another important annotation level. Such a segmentation of a spontaneous speech corpus is rather complex as we do not have access to punctuation. The corpus was therefore manually annotated to define segments with both a prosodic and a semantic completeness.

3. Prosody Annotation Protocol

We defined in [17] a two-level prosody annotation framework to consider both accentual and macro prosodic phenomena. The objective was to determine a specific set of labels for both annotation tiers. To allow for their integration in a speech synthesis system, the labels required to comply with two main constraints. First, they had to be related to a specific function to facilitate their prediction from text at synthesis stage. Secondly, they had to be characterized by distinct acoustic realizations. It should be noted that existing systems like ToBI [27] could hardly be exploited as such for our study as their complexity makes it difficult to predict the labels from the text.

Our local tier contains a small amount of labels (Table 1). Each label fulfills a distinct and specific function. Five labels are related to non-emphatic stresses [28] and are assigned to the end of accentual phrases. They are characterized by a pitch level, H for rising or higher pitch vs. L for falling or lower pitch. They can also be distinguished by the level of boundary they determine, similarly to boundary tones in [27]. To facilitate the automatic annotation of the labels (from the text or from the acoustics), these two levels are distinguished according to the presence or absence of a subsequent silence. Conversely to H and L syllables, HH and LL syllables are directly followed by a silent pause. A specific tag E is assigned to the final boundary of player names enumerations, which are very common in sports commentaries and may display a specific acoustic realization. A focus stress (F) relates to emphatic stresses. An hesitation label (He), and a creaky label (C) allow avoiding the degradation of the models at training time. Indeed, hesitations are realized with long durations whilst creaky syllables are characterized, among others, by a very low pitch [29]. If these syllables are not singled out, their prosodic features may influence the synthesized prosody. All remaining syllables are assigned a NA symbol.

Table 1: *List of local labels.*

Stresses			Unstressed	Other
Not emphatic		Emphatic		
H HH L LL E		F	NA	He C

The global tier is inspired by [15] and [16] and is assigned to groups of words, conversely to the local annotation which assigns a symbol to each syllable. It basically classifies the speech segments into speaking styles, based on a dimensional analysis of emotions [30]. The valence and the arousal levels drove us to define five speaking styles (Figure 1).

We showed in [17] that the different labels are, as required, associated with specific acoustic realizations. While 'F' labels tend to be realized with a higher pitch level but low syllable lengthening, non-emphatic stresses are usually characterized by an important lengthening of the syllable. Regarding global labels, the arousal level was shown to be correlated with the fundamental frequency, highly excited segments being realized with a significantly higher pitch. Inter-annotator rates were also computed. They reached a Cohen's kappa score [31] of 0.66 for the local labels, which is comparable to the rate obtained for ToBI [27]. The global annotation achieved lower rates but with logical interchanges between the labels [17].

4. Methodology and Experiments

In order to assess the validity of our local and global labels definition, several HMM-based speech synthesizers [5] were built,

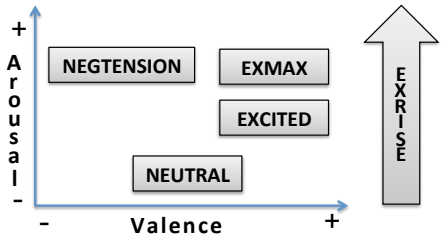


Figure 1: *The global labels on a dimensional scale.*

relying on the implementation of the HTS toolkit[1] (version 2.1) publicly available. For each synthesizer, 90% of the corresponding database was used for the training (called the *training set*), leaving around 10% for the synthesis (called the *synthesis set*). Note that both the training and synthesis sets were manually annotated with our two-layer prosody annotation. As filter parameterization, we extracted the Mel Generalized Cepstral (MGC) coefficients traditionally used in parametric synthesis. As excitation modeling, the Deterministic plus Stochastic Model (DSM [32]) of the residual signal was used to improve naturalness.

The influence of the integration of the local and global labels is first analyzed and quantified independently in Sections 4.1 and 4.2, respectively. Using the conclusions drawn from these latter evaluations, the combination of both local and global labels is studied in Section 4.3.

4.1. Integration of the Local Labels

This section is devoted to the integration of the single local annotation layer into HMM-based speech synthesis.

4.1.1. Method

The first synthesizer is trained on the entire training set of the database (Section 2), regardless of the speaking styles. This is our baseline system, called *Base*. The only contextual information provided during the training and synthesis stages is a manually-checked phonetic transcription, embedded as standard HTS labels [5].

The same training procedure is applied to the second synthesizer, called *Loc*. It makes use of the same phonetic transcription, but complemented in this case with specific contextual information from the local prosody annotation level (Table 1), replacing the unused ToBI field in the standard HTS labels.

4.1.2. Evaluation Protocol

A first Mean Opinion Score (MOS) test is conducted in order to quantify the impact of the local annotation layer in comparison with the baseline system. For this evaluation, participants were asked to listen to two versions of the same sentence synthesized by the two following models (randomly shuffled): (i) the baseline system (*Base*); (ii) the system integrating local labels (*Loc*). Each sentence was scored according to three criteria: intelligibility, expressivity and segmental quality. Listeners were given two continuous MOS scales (one for each criterion) ranging from 1 (meaning "poor") to 5 (meaning "excellent"). These scales were extended one point further on both sides (ranging therefore from 0 to 6) in order to prevent border effects.

[1] http://hts.sp.nitech.ac.jp/

The test consists of 10 pairwise comparisons. Sentences were randomly chosen amongst the synthesis set of the database. These sentences were not divided in speaking styles, which means that a sentence may correspond to a sequence of various speaking styles. 10 native French-speaking people, mainly naive listeners, participated in this evaluation. During the test, they could listen to the pair of sentences as many times as wanted in the order they preferred. They were nonetheless advised to first listen to the two sentences in a row so as to estimate approximately their relative position. However, they were not allowed to come back to previous sentences after validating their decisions.

4.1.3. Results

MOS scores are computed for all evaluations in this paper to provide comparable results in a coherent evaluation framework. The actual MOS scores are, however, less informative in this first evaluation. Therefore, our analysis relies on the preference percentages which are computed as the listener's relative preference for a synthesis method compared to another. Figure 2 shows the preference scores for the three criteria. The light grey segment corresponds to the proportion of cases in which both methods are assigned the same MOS score. It can be observed that *Loc* is preferred for the rendering of the expressivity. Interestingly, it is also shown to improve the segmental quality, while achieving an intelligibility level that is similar to the baseline (i.e. *Base*).

The analysis of the MOS scores further confirms that *Loc* allows to slightly increase the expressivity compared to *Base*. This means that local labels were properly learned during training of *Loc* and that specific accentual Probability Density Functions (PDFs) were properly estimated. At synthesis time, the model was thus able to predict more precise accentual realizations. On the contrary, since only a manually-checked phonetic transcription was provided for the *Base* training, all acoustic realizations that should have corresponded to local labels were merged into more global PDFs.

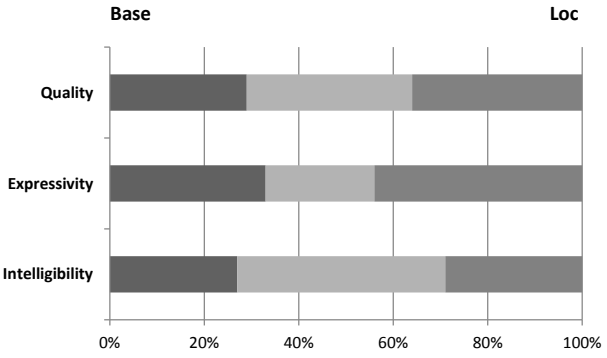


Figure 2: *Preference scores for each criterion and each synthesis method, i.e. with (Loc) and without (Base) the integration of the local labels.*

4.2. Integration of the Global Labels

This section studies the integration of the single global annotation layer into HMM-based speech synthesis. It should be noted that speaking styles defined by global annotation layer are not uniformly distributed throughout the corpus. The total duration of each speaking style of the database is shown in Table 2.

Table 2: *Total duration (in sec.) of the various speaking styles of our database, long silences (>1 sec.) being excluded.*

Excited	ExMax	ExRise	NegTension	Neutral
1032	475	485	582	2955

4.2.1. Method

Three distinct methods are investigated regarding the integration of the global labels in speech synthesis. The first method consists in training speaking style-dependent models on exclusive subsets of the whole corpus, specific to the global label they correspond to (see Table 2). They will be referred to as *full data models* in the remainder of the paper. At the end of this step, 5 different full data models, called *Glob1*, are obtained (i.e. Excited, ExMax, ExRise, NegTension and Neutral).

The two other methods exploit adaptation techniques. The second method relies on the fact that the Neutral style has the highest amount of speech data amongst the different speaking styles. Assuming that this amount of speech data is sufficient to obtain a strong Neutral *full data model*, voice adaptation techniques [18] can be applied to train more reliably the remaining models, for which less speech data are available. The *Glob1* Neutral *full data model* was then adapted using the Constrained Maximum Likelihood Linear Regression (CMLLR) transform [33] [34] in the framework of Hidden Semi Markov Model (HSMM) [35] with the adaptation sets of the four remaining speaking styles. It produces respectively Excited, ExMax, ExRise and NegTension HMM-based synthesizers. The linearly-transformed models were further optimized using MAP adaptation [18], providing the 4 adapted models called *Glob2*.

A potential drawback of the second method is that the speech data used to train the Neutral *full data model* may not be large enough. To alleviate this issue, Yamagishi proposed in [36] to adapt a so-called *average-voice model* to a particular target speaker. The average-voice model is computed once and for all over a database containing many different speakers. This technique proved to be efficient when few speech data is available. The average-voice model was here computed on the entire database, regardless of the speaking styles. This model was then adapted following the same procedure as for *Glob2*. We finally obtained 5 different adapted sub-synthesizers called *Glob3* (i.e. Excited, ExMax, ExRise, NegTension and Neutral).

4.2.2. Evaluation Protocol

A second MOS test is conducted in order to elect which of the three methods is the most suited for the integration of global labels. For this evaluation, participants were asked to listen to three versions of the same sentence, synthesized by the models corresponding to the three aforementioned methods. Each sentence was scored according to two criteria: intelligibility and segmental quality. The same experimental protocol as in Section 4.1.2 was applied. However, contrarily to that section, the expressivity was not assessed here. Indeed our first informal experiments showed that *Glob* models exhibit some intelligibility and segmental quality issues. These had to be addressed before focusing on a good rendering of the expressivity.

The test consists of 15 triplets. Sentences were randomly chosen amongst the synthesis set of the database. Conversely to Section 4.1.2, each sentence only contains one speaking style. 12 native French-speaking testers, mainly naive listeners, participated in this evaluation.

4.2.3. Results

For each synthesis technique and each speaking style, Figures 3 and 4 display respectively the averaged intelligibility and segmental quality MOS scores, together with their 95% confidence intervals (CI). It clearly turns out that *Glob3*, i.e. the adapted average models, provides the highest results, both in terms of intelligibility and segmental quality of the generated speech. The *full data models*, conversely, achieve the lowest scores in most speaking styles. As a reminder, a score of 3 or 4 on the MOS scale means respectively "Fair" or "Good".

This preference for the adapted average models can be explained by the fact that they are computed using all the training sets for each speaking style, thus providing a robust model which is then adapted to each speaking style. It can however be noted that Neutral, Excited and NegTension voices are better rendered than ExMax and ExRise ones. This is mainly due to the fact that Neutral, Excited and NegTension have more speech data, leading to a better average-voice model adaptation compared to ExMax and ExRise.

It should also be noted that all the synthesizers achieved the same performance for the Neutral speaking style. This can be understood by the fact that this latter style is the only one having a comfortable amount of speech data for a reliable estimation of the model, independently of the training method.

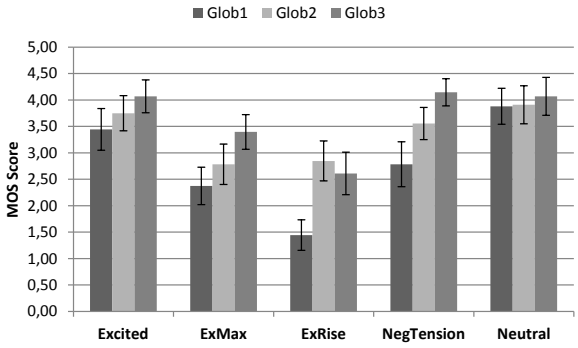


Figure 3: *Averaged **intelligibility** MOS scores for each synthesis method and each speaking style, with their 95% CI.*

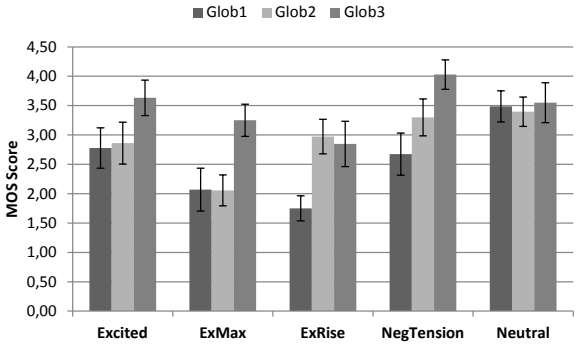


Figure 4: *Averaged **segmental quality** MOS scores for each synthesis method and each speaking style, with their 95% CI.*

Preference scores corroborate these findings. Regarding intelligibility, *Glob3* is respectively preferred in 69.4% and 51.1% of the cases compared to *Glob1* and *Glob2* (which are assigned 13.3% and 25.6% of the preferences). *Glob3* is chosen in 73.9%

and 64.4% of the cases over respectively *Glob1* and *Glob2* for its segmental quality.

This shows that, as the amount of speech data is unevenly distributed amongst the different speaking styles, adapting a robustly trained average-voice model with an efficient technique such as CMLLR allows generating various speaking styles of reasonable intelligibility and segmental quality.

4.2.4. Comparison with the Baseline

Similarly to the integration of local labels (Section 4.1.3), the integration of global labels was compared to the baseline through a MOS test. For this comparison, the best integration technique, i.e. the adaptation of the average model, was used. The baseline is the same as in Section 4.1.3, which means that it disregards both annotation layers. 20 native French-speaking people, mainly naive listeners, participated in this evaluation. Conversely to what was expected, no statistically significant differences were observed between both methods. This integration achieves indeed comparable or even slightly lower scores in terms of intelligibility, expressivity and segmental quality.

4.3. Integration of both Local and Global Labels

This section is devoted to the integration of the two-layer annotation (both local and global labels) into HMM-based synthesis.

4.3.1. Method

We showed in Section 4.1.3 that the integration of local labels results in an enhanced expressivity, while it provides slightly higher intelligibility and segmental quality performance. Regarding the integration of global labels, the use of adaptation techniques, from an average model, was shown to provide the best results (see Section 4.2.3). However, this annotation level seemed to achieve no improvement regarding expressivity in comparison with a baseline model.

We investigate, in this last section, whether the integration of both local and global labels achieves higher scores. These combined models are referred to as *Loc+Glob3*.

4.3.2. Evaluation Protocol

A third MOS test is conducted. For this evaluation, participants were asked to listen to four versions of the same sentence synthesized by the following models (randomly shuffled): (i) the baseline model (*Base*); (ii) the model integrating local labels (*Loc*); (iii) the model adapted from the average-voice model (*Glob3*); (iv) the model adapted from the average-voice model integrating local labels (*Loc+Glob3*). Here again, intelligibility, expressivity and segmental quality are evaluated. The same experimental protocol as in Section 4.1.2 was applied.

The test consists of 10 quadruplets. Sentences were randomly chosen amongst the synthesis sets of each speaking style of the database. Similarly to Section 4.2.2, each sentence only contains one speaking style. 20 native French-speaking people, mainly naive listeners, participated in this evaluation.

4.3.3. Results

Table 3 shows the preference scores for the four methods. It should be noted that the scores obtained by two reverse pairs are not summing to 100%. This is due to the fact that cases when both methods composing the considered pair are found to be equivalent are also taken into account. Regarding intelligibility, Table 3 shows for example that *Base* is preferred to *Loc* in 31% of the cases, while *Loc* is preferred to *Base* in 35.5% of the cases. The remaining percentage, i.e. 33.5%, corresponds then to the cases where both *Base* and *Loc* are equivalently preferred.

As in Section 4.1.3, it is observed that the integration of the local labels carries out an improvement in the rendering of the expressivity and provides comparable or slightly better intelligibility and segmental quality of the generated speech. On the other hand, the integration of the global labels only (using the adapted average models) does not improve any of the analyzed criteria compared to the baseline, which corroborates the results obtained in Section 4.2.4. Regarding the integration of both prosody annotation levels, an insightful observation is that *Loc+Glob3* is preferred in 41% of the cases in terms of expressivity against *Loc*, which is assigned 35% of the preferences. The segmental quality degrades, however, from *Loc* to *Loc+Glob3* as they are respectively preferred in 39.5% and 31.5% of the time. Nonetheless, both methods achieve similar intelligibility performance.

Table 3: *Integration of both Global and Local Labels - Preference scores (in [%]) for each method and each criterion.*

		Base	*Loc*	*Glob3*	*Loc+ Glob3*
Intelligibility	*Base*	0	31	33	29.5
	Loc	35.5	0	37.5	29.5
	Glob3	29	29	0	26.5
	Loc+Glob3	40	27.5	38	0
Expressivity	*Base*	0	37.5	40	32.5
	Loc	45.5	0	46	35
	Glob3	35.5	35	0	29.5
	Loc+Glob3	51	41	47.5	0
Quality	*Base*	0	39.5	40	44
	Loc	38.5	0	46.5	39.5
	Glob3	27.5	33.5	0	35
	Loc+Glob3	36	31.5	45.5	0

5. Conclusion

In this paper, we proposed the integration of a two-layer prosody annotation specific to live sports commentaries into HMM-based speech synthesis. The local annotation relates to accentual phenomena while the global layer classifies the speech segments into distinct speaking styles. Our study was divided into three parts.

First, the improvement carried out by local labels was quantified by comparing: (i) a baseline model, in which a manually-checked phonetic transcription was the only contextual information provided and (ii) a model integrating local labels. Subjective tests revealed that, compared to the baseline, the integration of local labels results in an enhanced expressivity, while providing slightly higher intelligibility and segmental quality scores.

Secondly, the integration of global labels (i.e. speaking styles) was evaluated. Three methods were investigated: (i) a speaking style-dependent training and the adaptation of (ii) the neutral model or (iii) the average-voice model to each speaking style. It was shown that adaptation techniques, and the adaptation from an average-voice model in particular, outperform style-dependent models both in terms of intelligibility and segmental quality. However, the comparison with the baseline, i.e. the model disregarding global labels, showed that, contrary to

what was expected, the integration of global labels does not enhance expressivity and slightly degrades the segmental quality.

A last experiment allowed evaluating the effects achieved by the combination of both prosody annotation layers on the generated speech. Interestingly, the complete integration of the two-layer annotation, compared to the model integrating local labels only, led to an even better rendering of expressivity, while achieving similar intelligibility scores. However, it slightly degrades the segmental quality. Our future work should thus focus on the improvement of speaking style adaptation techniques in order to increase the segmental quality of the generated speech.

Audio examples related to this study are available online at http://tcts.fpms.ac.be/~picart/.

6. Ackowledgements

Authors are supported by FNRS. The project is partly funded by the Walloon Region Wist 3 SPORTIC. Authors are grateful to S. Audrit for her implication in the recording of the corpus.

7. References

[1] R. Tsuzuki, H. Zen, K. Tokuda, T. Kitamura, M. Bulut, and S. Narayanan, "Constructing emotional speech synthesizers with limited speech database," in *International Conference on Spoken Language Processing (ICSLP)*, 2004, pp. 1185–1188.

[2] J. Yamagishi, K. Onishi, T. Musuko, and T. Kobayashi, "Acoustic modeling of speaking styles and emotional expressions in hmm-based speech synthesis," *IECE Transactions on Information and Systems*, vol. E88-D(3), pp. 502–509, 2005.

[3] T. Takahashi, T. Fujii, M. Nishi, H. Banno, T. Irino, and H. Kawahara, "Voice and emotional expression transformation based on statistics of vowel parameters in an emotional speech database," in *Interspeech*, 2005, pp. 537–540.

[4] L. Qin, Z.-H. Ling, Y.-J. Wu, B.-F. Zhang, and R.-H. Wang, "HMM-based emotional speech synthesis using average emotion models," in *ICSLP*, 2006, pp. 233–240.

[5] H. Zen, K. Tokuda, and A. Black, "Statistical parametric speech synthesis," *Speech Comm.*, vol. 51(11), pp. 1039–1064, 2009.

[6] T. Yoshimura, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, "Simultaneous modeling of spectrum, pitch and duration in HMM-based speech synthesis," in *Eurospeech*, 1999, pp. 2347–2350.

[7] I. Fonagy, *L'accent en français contemporain*. Ottawa: Marcel Didier Ltée, 1979, ch. L'accent français : Accent probabilitaire, pp. 123–232.

[8] K. Hirose, K. Sato, and N. Minematsu, "Emotional speech synthesis with corpus-based generation of f0 contours using generation process model," in *Speech Prosody*, 2004, pp. 417–420.

[9] M. E. Beckman and J. B. Pierrehumbert, "Japanese prosodic phrasing and intonation synthesis," in *Twenty-Fourth Annual Meeting of ACL*, 1986, p. 173180.

[10] N. Braunschweiler, M. J. Gales, and S. Buchholz, "Lightly supervised recognition for automatic alignment of large coherent speech recordings," in *Interspeech*, 2010, pp. 2222–2225.

[11] F. Eyben, S. Bucholz, and N. Braunschweiler, "Unsupervised clustering of emotion and voice styles for expressive TTS," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2012.

[12] S. Audrit, T. Psir, A. Auchlin, and J.-P. Goldman, "Sport in the media: A contrasted study of three sport live media reports with semi-automatic tools," in *Speech Prosody*, 2012.

[13] J. Trouvain and W. Barry, "The prosody of excitement in horse race commentaries," in *ISCA Workshop on Speech and Emotion: A Conceptual Framework for Research*, 2000, pp. 86–91.

[14] N. Obin, V. Dellwo, A. Lacheret, and X. Rodet, "Expectations for discourse genre identification," in *Interspeech*, 2010.

[15] J. Trouvain, "Between excitement and triumph - live football commentaries in radio vs. tv," in *17th International Congress of Phonetic Sciences (ICPhS XVII)*, 2011.

[16] F. Kern, *Prosody in Interaction*. John Benjamins, 2010, ch. Speaking Dramatically. The Prosody of Live Radio Commentary of Football Matches, pp. 217–237.

[17] S. Brognaux, B. Picart, and T. Drugman, "A new prosody annotation protocol for live sports commentaries," in *Interspeech*, 2013.

[18] J. Yamagishi, T. Nose, H. Zen, Z. Ling, T. Toda, K. Tokuda, S. King, and S. Renals, "Robust speaker-adaptive hmm-based text-to-speech synthesis," *IEEE Audio, Speech, & Language Processing*, vol. 17(6), pp. 1208–1230, 2009.

[19] J. Yamagishi, T. Masuko, and T. Kobayashi, "Hmm-based expressive speech synthesis - towards tts with arbitrary speaking styles and emotions," in *Proc. of SWIM*, 2004.

[20] T. Nose, M. Tachibana, and T. Kobayash, "Hmm-based style control for expressive speech synthesis with arbitrary speakers voice using model adaptation," *IEICE Transactions on Information and Systems*, vol. 92(3), pp. 489–497, 2009.

[21] J. Yamagishi, "Average-voice-based speech synthesis," Ph.D. dissertation, Tokyo Institute of Technology, 2006.

[22] H. Zen, N. Braunschweiler, S. Buchholz, M. J. Gales, K. Knill, S. Krstulovic, and J. Latorre, "Statistical parametric speech synthesis based on speaker and language factorization," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 20(6), pp. 1713–1724, 2012.

[23] N. Obin, P. Lanchantin, A. Lacheret, and X. Rodet, "Discrete/continuous modelling of speaking style in hmm-based speech synthesis: Design and evaluation," in *Interspeech*, 2011.

[24] J.-P. Goldman, "Easyalign: an automatic phonetic alignment tool under Praat," in *Interspeech*, 2011, pp. 3233–3236.

[25] S. Brognaux, S. Roekhaut, T. Drugman, and R. Beaufort, "Train&Align: A new online tool for automatic phonetic alignments," in *IEEE SLT Workshop*, 2012.

[26] V. Colotte and R. Beaufort, "Linguistic features weighting for a text-to-speech system without prosody model," in *Interspeech*, 2005, pp. 2549–2552.

[27] K. Silverman, M. Beckman, J. Pitrelli, M. Ostendorf, C. Wightman, P. Price, J. Pierrehumbert, and J. Hirschberg, "ToBI: A standard for labeling english prosody," in *ICSLP*, 1992, pp. 867–870.

[28] A. Di Cristo, "Vers une modélisation de l'accentuation du français: deuxième partie," *Journal of French Studies*, vol. 10, pp. 27–44, 2000.

[29] T. Drugman, J. Kane, and C. Gobl, "Modeling the creaky excitation for parametric speech synthesis," in *Interspeech*, 2012.

[30] A. Mehrabian and J. A. Russel, *An Approach to Environmental Psychology*. MIT Press, 1974.

[31] J. Cohen, "A coefficient of agreement for nominal scales," *Educational and Psychological Measurement*, vol. 20(1), pp. 37–46, 1960.

[32] T. Drugman and T. Dutoit, "The deterministic plus stochastic model of the residual signal and its applications," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 20(3), pp. 968–981, 2012.

[33] V. Digalakis, D. Rtischev, and L. Neumeyer, "Speaker adaptation using constrainted reestimation of gaussian mixtures," *IEEE Transactions on Speech and Audio Processing*, vol. 3(5), pp. 357–366, 1995.

[34] M. Gales, "Maximum likelihood linear transformations for hmm-base speech recognition," *Computer Speech and Language*, vol. 12(2), pp. 75–98, 1998.

[35] J. Ferguson, "Variable duration models for speech," in *Symp. on Application of Hidden Markov Models to Text and Speech*, 1980.

[36] J. Yamagishi and T. Kobayashi, "Average-voice-based speech synthesis using hsmm-based speaker adaptation and adaptive training," *IEICE Transactions Information and Systems*, vol. 90(2), pp. 533–543, 2007.

Parametric model for vocal effort interpolation with Harmonics Plus Noise Models

Àngel Calzada Defez[1] *, Joan Claudi Socoró Carrié*[1]*, Robert A. J. Clark*[2]

[1]Human Computer Interaction Department,
Enginyeria i arquitectura La Salle (Universitat Ramon Llull), Barcelona, Spain
[2] The Centre for Speech technology Research, The University of Edinburgh, U.K.
acalzada@salle.url.edu, jclaudi@salle.url.edu, robert@cstr.ed.ac.uk

Abstract

It is known that voice quality plays an important role in expressive speech. In this paper, we present a methodology for modifying vocal effort level, which can be applied by text-to-speech (TTS) systems to provide the flexibility needed to improve the naturalness of synthesized speech. This extends previous work using low order Linear Prediction Coefficients (LPC) where the flexibility was constrained by the amount of vocal effort levels available in the corpora. The proposed methodology overcomes these limitations by replacing the low order LPC by ninth order polynomials to allow not only vocal effort to be modified towards the available templates, but also to allow the generation of intermediate vocal effort levels between levels available in training data. This flexibility comes from the combination of Harmonics plus Noise Models and using a parametric model to represent the spectral envelope. The conducted perceptual tests demonstrate the effectiveness of the proposed technique in performing vocal effort interpolations while maintaining the signal quality in the final synthesis. The proposed technique can be used in unit-selection TTS systems to reduce corpus size while increasing its flexibility, and the techniques could potentially be employed by HMM based speech synthesis systems if appropriate acoustic features are being used.

Index Terms: vocal effort interpolation, harmonics plus noise model, expressive speech synthesis

1. Introduction

Technology is currently embedded in society; however, some barriers, such as deficiencies in natural speech in *Human-Machine-Interfaces* (HMI) remain, thus preventing technology from reaching several communities. In these cases, improved *Text-To-Speech* (TTS) systems can make the HMI more natural improving the user experience when using technology [1].

The *Harmonics Plus Noise Model* (HNM) allows us to easily perform prosody modifications on speech signals, while maintaining a high level of quality in the resulting synthesized signal [2]. For this reason, this model has been chosen as a speech signal representation model by speech modification applications aiming to improve the naturalness and expressiveness of the TTS system. The HNM has also been used in expressive speech synthesis systems where both prosody and voice quality (VoQ) were modified [3, 4, 5, 6]. It has been proven that VoQ has an important role in expressive speech synthesis [7, 8], which led us to speculate whether the HNM could be used to modify low-level VoQ parameters [5]. This work [5], asserted the relevance of VoQ in the expressive style perceived by the listener and confirmed the feasibility of using the HNM to modify VoQ parameters. Despite having achieved good results in terms of expressiveness in [5], the quality of the synthesized signal was seriously degraded. The number of signal manipulations[1] and the unresolved existing interdependence of some spectral parameters (i.e., the Hammarberg Index -hammi- and the relative amount of energy above 1000 Hz -pe1000-) caused degradation of the synthesized signal's quality. To simplify the procedure and focus on high-quality modifications, the number of parameters modified was reduced to just one, vocal effort; this was chosen for its salient role in expressive speech characterization [9]. The proposed methodology in [10] proved the feasibility of HNM for modifying vocal effort with a model based on low order linear prediction coefficients (LPC). However, the method based on low order LPC is very sensitive to interpolation artefacts which might lead filter instability. Thus, in order to cover multiple levels of vocal effort, it requires data for each target vocal effort level to to be able to synthesize with that level of vocal effort. This makes the methods flexibility dependent upon corpus size.

In this work, we present a new model based on ninth order polynomials for representing the harmonic spectral envelope which not only allows the transfer of vocal effort from a template signal available in the corpus, but also allows us to generate intermediate vocal effort levels not present in the available corpus. The proposed methodology can be applied in the context of TTS systems, particularly to allow them to synthesize speech signals expressing a range of vocal effort levels. Moreover, vocal effort levels that are different to those present in the original corpora recordings that a voice is based upon.

This paper is organized as follows. Section 2.1 reviews the details of the implementation of the HNM speech signal parameterization and re-synthesis. The speech database used for the experiments is briefly explained in section 2.2 and section 2.3 presents the polynomial model and details how the model codebooks were built from the original corpus. Next, the proposed vocal effort modification procedure is detailed in section 2.4. In section 3, the conducted perceptual experiments are outlined. Section 4 discusses the proposed procedure, the obtained results and future work. Fianlly section 5 provides conclusions.

[1]Up to five parameters (jitter, shimmer, hnr, Hammarberg Index -hammi- and the relative amount of energy above 1000 Hz -pe1000-) were modified based on a five-stage procedure, in which each stage was specifically designed to modify a unique VoQ parameter.

2. Methods

2.1. Harmonics Plus Noise Model

The HNM expresses the sampled speech signal $s[n]$ as the sum of two components: $h[n]$ and $r[n]$, which correspond to the harmonic and noise, or stochastic, components of the signal, respectively.

$$s[n] = h[n] + r[n] \quad (1)$$

The harmonic component, $h[n]$, models the pseudo-periodicity that appears in the speech signal as a sum of harmonically related sinusoids. Given a quasi-periodic frame k with L_k harmonics, the harmonic part is characterized by the amplitude[2] ($\mathbf{A}_k$), frequency[3] ($\mathbf{F}_k$) and phase ($\mathbf{\Phi}_k$) arrays. In contrast, the stochastic component, $r[n]$, models all non-periodic events in the speech signal with an autoregressive (AR) model and is represented through Q-order Linear Prediction Coefficients (LPC) and noise variances (P_k). From the frequency-domain perspective, the harmonic component mainly models the lower band of the spectrum, whereas the upper band is modeled by the stochastic component. The boundary between these two bands is known as the maximum voiced frequency (f_h). Some authors consider this boundary variable in time [11], but the HNM implementation used in the present work fixed this boundary to 5 kHz, as in [12, 13]. All signals have been analyzed at a constant frame rate of 5*ms* for both, harmonic and stochastic parts.

2.2. Speech database

The speech material used to validate the proposed vocal effort transformation methodology was the German diphone set recorded with three degrees of vocal effort (vocal effort levels were labelled as *Low*, *Modal* and *High*), as reported in [9]. The corpus was divided into six datasets containing logatoms of three syllables in length with voiced and unvoiced sounds recorded with a constant pitch. Recordings of the three vocal effort levels from a male and female speaker were available. As explained in [9], the data were automatically labelled and subsequently hand-corrected.

2.3. Parametric model and code-books

The entire corpus was represented using HNM parameters. Informal tests conducted prior to the proper tests presented in this work, highlighted some speech signal variations in the logatome syllables due to syllable position inside the utterance. For the sake of obtaining more accurate models without losing too much generalization, the syllable position inside the logatome were considered together with phoneme labels for indexing the models in the code-books. Therefore, each dataset (corresponding to a specific vocal effort level and gender) was divided into three sub-datasets, one per each syllable position in the logatome (*init, middle* and *final*). For each sub-dataset all HNM parameters from multiple realizations of a common phoneme were gathered together. Only parameters from the stable part of the phonemes were used to prevent coarticulation effects from being introduced in the final computed models. The stable part was considered to be the second and third quartiles of the full phoneme duration (figure 2). At this point we had the HNM parameters of all realizations of the same phoneme for a given vocal effort level, gender and syllable position in the logatome together. Next, the parametric model coefficients were computed to fit all the data points formed by the harmonic amplitudes and frequencies for a given phoneme. Finally, the model parameters for all sub-data sets corresponding to the same gender and vocal effort level were gathered together forming the code-book for that vocal effort level and gender. Up to six code-books were generated covering all vocal effort levels and gender combinations. For a given synthesis, only the three code-books from the gender to synthesize are used for carrying out the vocal effort modification and synthesis process.

Code-books are used to retrieve extreme vocal effort models, which in our case are labelled as *High* and *Low*. The third code-book labelled as *Modal* is used as the reference level. Apart from the extreme *Low* and *High* vocal effort levels, the proposed methodology aims to allow the TTS system to also perform intermediate vocal effort levels. Polynomials were chosen in order to be able to interpolate between models from the code-books. The proposed methodology uses ninth order polynomials (eq. (2)) to be able to capture the fourth formant peaks and valleys. Informal tests were conducted computing the general root-mean squared error RMSE for different number of coefficients. The informal tests showed that the main error reduction was achieved with the first five coefficients.

$$\widehat{ampl}(f) = a_0 + a_1 f + a_2 f^2 + a_3 f^3 + \cdots + a_9 f^9 \quad (2)$$

where $\widehat{ampl}(f)$ is the harmonic's amplitude envelope which is a function of the harmonic's frequencies f, and a_i for $i \in [0, 9]$ are the model coefficients. Table 1 presents an excerpt from the code-book corresponding to the *High* vocal effort level for the female speaker.

Table 1: *Sample of two phonemes information taken from the code-book corresponding to High vocal effort level from the female speaker. Phoneme labels follow the SAMPA notation[14]. Part of the coefficients have been removed in order to fit the table width to the column dimensions.*

Phoneme	Syllabe position	Model coefficients
...	...	...
U;	init;	1.941e-31;-4.810e-27;$\cdots$;-0.067;
U;	middle;	1.501e-31;-3.761e-27;$\cdots$;-0.039;
U;	final;	1.071e-31;-2.664e-27;$\cdots$;-0.030;
o;	init;	2.046e-31;-5.298e-27;$\cdots$;-0.118;
o;	middle;	3.044e-32;-9.366e-28;$\cdots$;-0.040;
o;	final;	-1.502e-31;3.287e-27;$\cdots$;0.045;
...	...	...

2.4. Proposed methodology for applying polynomial models for vocal effort modification

The proposed methodology uses the modal vocal effort level data as the starting point for the modifications. This decision was based on the results obtained in previous work [10] where it was found that the signal quality degradation was directly related with the amount of signal modification. Thus, in order to minimize the amount of signal modification for all cases, rising and lowering the vocal effort level, we decided to use modal vocal effort level as the source for all the signal modifications.

[2] $\mathbf{A}_k = \{A_k^1, A_k^2, \cdots, A_k^l\}$, where the super-index l indicates the harmonic number $l \in [1, .., L_k]$.

[3] $\mathbf{F}_k = \{F_k^1, \cdots, F_k^l\}$. Frequencies are harmonic; therefore, $F_k^l = l\ F_0(k)$, where $F_0(k)$ corresponds to the fundamental, or pitch, frequency for a given frame, k.

For this reason only the HNM parameters from the modal corpus will be used for synthesis. HNM parameters from high and low datasets are only used for building the respective (*High* and *Low*) code-books. Figure 1 depicts the general schema for the proposed methodology.

The vocal effort synthesis procedure conducted in this work begins with a given transcription of the text to be synthesized. The transcription is used to retrieve the corresponding model parameters from the code-books. The modal code-book is always accessed because the spectral envelope from this vocal effort level will be used as the baseline for the posterior modifications. However, the *High* and *Low* code-books are used only when necessary. The decision is taken based on the target vocal effort to be synthesized. The direction of the vocal effort modification is encoded in the sign of the *interpolation factor* (γ). In our case this factor is introduced with a real value from the range $[-1, 1]$. Negative values correspond to lowering the vocal effort, whereas positive values are used for increasing it. Thus, the extreme values (-1 and 1) indicate using the *Low* and *High* vocal effort parameter models as retrieved from the corresponding code-book.

Once the identified code-books to be used are loaded, the transcription is divided into three regions, where each region corresponds to a syllable from the logatome (regions were labelled as: *init, middle* and *final*). This information is used in combination with the phoneme label for searching the model units in the code-books. For instance, given the following transcription: $/t-a-m-u-t-a/$ the first unit to search will be the phoneme $/t/$ with an indicator of initial (*init*) position. However, the model parameters for the second $/t/$ will be different due to its different position in the logatome (*final*). Once the proper units for the whole sentence are selected from the corresponding code-books, the model coefficients are linearly interpolated in order to have model parameters for each frame to synthesize. However, the linear interpolation is carried out only in the unstable parts of the phonemes, where coarticulation effects are present. For the central regions of the phoneme and the beginning and end of the utterance to synthesize the original model parameters obtained from the code-book are replicated (figure 2). This process results in two matrices $\boldsymbol{Ec}$ and $\boldsymbol{Mc}$ with dimensions (mxn), where m corresponds to the number of coefficients in the model and n is the number of frames. Thus, the matrices contain the model parameters for each frame. $\boldsymbol{Ec}$ contains the extreme vocal effort parameters obtained from the *Low* or *High* vocal effort code-books, depending on γ sign, and $\boldsymbol{Mc}$ contains the modal vocal effort model parameter values for each frame.

The next step is to obtain the matrix corresponding to the interpolated vocal effort level ($\boldsymbol{I_c}$) from $\boldsymbol{Ec}$ and $\boldsymbol{Mc}$. Equation (3) shows the expression for computing the interpolated model parameters for a general frame k.

$$\boldsymbol{I_{c_k}} = (1 - |\gamma|)\, \boldsymbol{M_{c_k}} + |\gamma|\ \boldsymbol{E_{c_k}};\ \ \gamma \in [-1, 1] \qquad (3)$$

where $\boldsymbol{I_{c_k}}$ are the interpolated model coefficients corresponding to the final desired vocal effort level for the k^{th} frame, γ is the interpolation factor and $\boldsymbol{M_{c_k}}$ and $\boldsymbol{E_{c_k}}$ are the coefficients from k^{th} frame for the Modal and extreme (*Low* or *High* depending on $sign(\gamma)$) vocal effort levels respectively.

Once the final desired model coefficients are computed, the models ($\boldsymbol{I_c}$ and $\boldsymbol{M_c}$) are evaluated at the original signal's harmonic frequencies ($\boldsymbol{F}$) using the expression (2) to obtain the harmonic spectral envelopes. The harmonic spectral envelope is

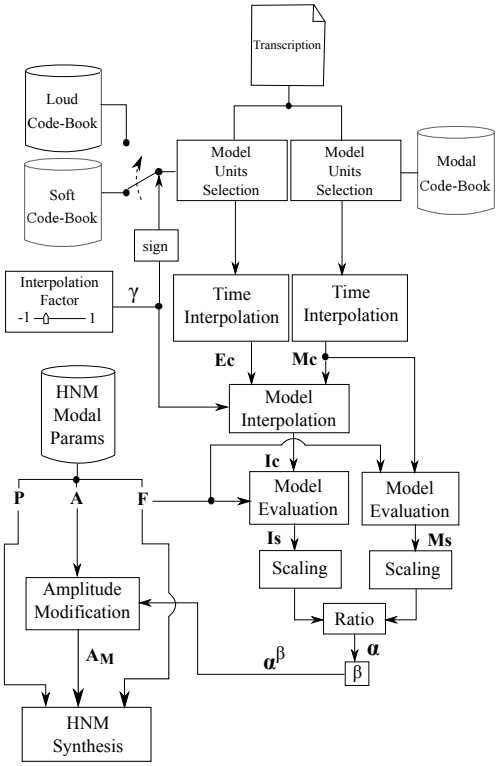


Figure 1: *Schematic diagram of vocal effort interpolation method proposed.*

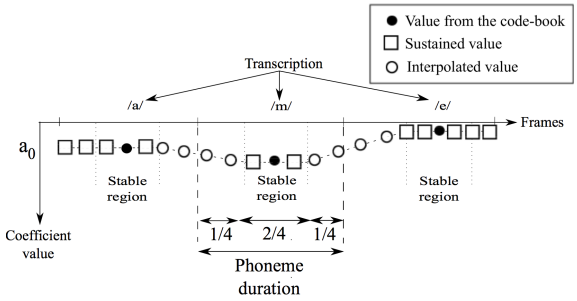


Figure 2: *Temporal linear interpolation of model coefficients.*

computed for both, the modal ($\boldsymbol{M_c}$) and the desired vocal effort level ($\boldsymbol{I_c}$) coefficient matrices obtaining $\boldsymbol{M_s}$ and $\boldsymbol{I_s}$ respectively. The multiplication factors ($\boldsymbol{\alpha}$) that will be applied to the HNM amplitudes ($\boldsymbol{A}$) from the original signal will be obtained from the harmonic spectral envelopes quotient for each frame.

$$\boldsymbol{\alpha_k} = \frac{\boldsymbol{I_{s_k}}}{\boldsymbol{M_{s_k}}} \qquad (4)$$

However, the magnitude of the harmonic spectral envelopes might contain values close to zero which could induce to numerical instabilities. In order to prevent this effect and focus on the energy distribution over the spectrum, the envelopes are scaled to fit into the range $[1, 2]$ prior to carrying out the ratio for computing the multiplicative factors ($\boldsymbol{\alpha}$).

In order to emphasize the effect of the energy distribution, the factors are powered to a factor β. This factor depends on the desired magnitude of the signal modification. Powering the factors to β amplifies the difference between peaks and valleys in

Table 2: SAMPA [14] transcription of the logatoms used in the first test.

[	aI	-	t - a:	-	p - aI	]
[	t - a:	-	f - u:	-	f - a:	]
[	t - a:	-	m - U	-	m - a:	]

the harmonic spectral envelope after the modification is applied.

Next the HNM amplitudes ($\boldsymbol{A}$) are modified applying the powered factors ($\boldsymbol{\alpha}^{\beta}$) obtaining the modified amplitudes ($\boldsymbol{A_M}$). Finally, the energy of each frame is adjusted in order to maintain the original energy magnitude that the frame had before applying any signal modification.

Last step is synthesizing the signal with regular HNM resynthesis procedure using the original frequencies ($\boldsymbol{F}$) and phases ($\boldsymbol{P}$), and the modified amplitudes ($\boldsymbol{A_M}$).

3. Results

Two perceptual tests were designed with the online testing platform for multimedia evaluation (TRUE) [15]. The first test with 22 users was focused on comparing the overall quality of the proposed method using ninth order polynomials against the previous proposal using low order LPC [10]. The second test with 21 users evaluated the performance of the proposed method interpolating vocal effort levels between the extreme levels available in the corpora (*Low* and *High*). At the beginning of each test, the users were presented a set of sample audio files expressing several vocal effort levels. In both tests the users were forced to decide between two answers (*A* or *B*). In order to prevent introducing any bias in the users' answers, for each pair of audio files evaluated, their labels (*A* or *B*) were randomly assigned. All statistical significance (*p-values*) have been computed using one-tailed test. The values used in the tests for the β parameter, used for powering the multiplicative factors ($\boldsymbol{\alpha}$), were set according to informal evaluations carried out prior to building the audio samples used in the conducted tests presented in this study. Thus, for the male gender β was set to 10, whereas for the female speaker it was set to 7.

The first test consisted of 3 logatoms (see table 2) uttered by both genders. Each logatom was applied two vocal effort conversions, from modal to low (*M2L*) and modal to high (*M2H*). The conversions were carried out by each one of the two methods under evaluation. Thus, the test was compressed by a total of 16 audio samples presented to the user in pairs. For a given pair of samples, both samples corresponded to the same vocal effort conversion performed by each one of the two methods. The user was asked to answer the following two questions for each pair:

1. Omitting the signal quality, which of the following files -A or B- performs a **Higher Vocal Effort?**
2. Which of the following samples -A or B- has better signal quality?

Table 3 presents the results of the first test where the performance of the proposed method was compared with the previous proposal [10]. The results show global preference for the new proposed method based on polynomial models interpolation. Regarding vocal effort modification there is a $53,79\%$ preference whereas in terms of signal quality this preference is more accentuated reaching a 82.95%. For obtaining the p-values of the results the null hypothesis (H_0) was set to: *There is no preference between the proposed method and the reference method*

Table 3: *Method preference according to vocal effort performance in test 1. P-value has been computed considering no preference between the methods as the null hypothesis (H_0)*

Parameter evaluated	Preference of proposed method [%]	P-value
Vocal effort conversion	53.7879	0.1093
Converted signal quality	82.9545	< 0.0001

[10]. The obtained p-values (*p-value* $= 0.1093$) state that in regards to vocal effort modification, there is no strong evidence for the proposed methodology. On the other hand, regarding to signal quality, the preference for the proposed methodology is statistically significant (*p-value* < 0.0001). Statistics have been computed using one-tailed significance tests to the sampling distribution.

With the result obtained from the first test, we conclude that despite not presenting relevant improvement for extreme vocal effort modifications when compared with the previous approach [10], it performed better in terms of signal quality. This result proves the suitability of the proposed method for transferring vocal effort.

The purpose of the second test was to verify the feasibility of using polynomial models to interpolate vocal effort levels. In order to prove its flexibility, several vocal effort levels were generated from the same modal vocal effort level utterance. Thus, using the modal (*M*) vocal effort level as a reference the following four vocal effort levels were synthesized: low (*L*), intermediate low (*IL*), intermediate high (*IH*) and high (*H*). Samples labelled as *IH* correspond to a linear interpolation of vocal effort levels high (*H*) and modal (*M*) with interpolation factor $\gamma = 0.5$ using the expression (3). Thus, samples labeled as *IH* were expected to be perceived between high (*H*) and modal (*M*) vocal efforts. On the other hand, samples labelled as *IL*, correspond a linear interpolation between low (*L*) and modal (*M*) vocal effort levels with an interpolation factor of $\gamma = -0.5$ applying equation (3). Likewise, samples labelled as *IL* were expected to be perceived as being between modal (*M*) and low (*L*) vocal effort levels.

The perceived vocal effort level, for each synthesized sample, was compared with the samples corresponding to the surrounding vocal effort levels. Extreme vocal effort levels were also compared with the modal reference. Thus, the users evaluated the following vocal effort level pairs: *L-IL*, *IL-M*, *M-IH*, *IH-H*, *L-M* and *M-H*. Extreme vocal effort levels *L* and *H* were synthesized using the models from their respective code-books using $\gamma = -1$, for *L*, and $\gamma = 1$ for *H*. The question for each pair of samples was: *Which of the following files -A or B- performs a **Higher Vocal Effort?***. Users were forced to choose between one of the two samples. Each pair presented to the user corresponded to two vocal effort levels synthesized from the same modal reference utterance for the same gender. The pairs presented to the user were randomized to prevent biases in the answers.

Three logatomes were taken from each gender obtaining the six different utterances used for the second test (see table 4). Each user evaluated each conversion six times, adding up a total of 132 evaluations for each vocal effort level comparison.

Tables 5 and 6 presents the results from the second test which evaluated the interpolation of vocal effort levels of the proposed methodology. Table 5 presents the results from the comparison of the synthesized versions for high (*H*) and low

Table 4: SAMPA [14] transcription of the logatoms used for the second test.

[	t - a:	-	s - i:	-	s - a:]
[	t - a:	-	j - a	-	j - a:]
[	t - a:	-	l - i:	-	l - a:]
[	t - a:	-	t - o:	-	t - a:]
[	t - a:	-	r - @:	-	r - a:]
[	t - a:	-	p - Y	-	p - a:]

Table 5: *Perception of extreme vocal efforts synthesized with the modal vocal effort level. The null hypothesis (H_0) was considered that users couldn't perceive any vocal effort level difference between each pair of samples.*

VE level pair	[%]	P-value
$H > M$	84.0909	< 0.01
$M > L$	90.9091	< 0.01

(*L*) vocal effort levels with the modal (*M*) version. Results state the general ordering for low (*L*), modal (*M*) and high (*H*) vocal effort levels. This results prove that samples synthesized with low vocal effort (*L*) are perceived as expected compared against modal (*M*), whereas those samples synthesized with high vocal effort (*H*) level are also perceived as expected when compared against the modal (*M*) reference.

Thus, results from table 5, prove that users perceived the synthesized extreme vocal effort levels according to the following ordering: $L < M < H$. The analysis of interpolated vocal effort levels (*IL* and *IH*) can be found in table 6.

As can be seen in table 6, *IL* synthesized samples, which are supposed to represent vocal effort levels between modal and low, were perceived as expected. When comparing *IL* with *M*, the success rate was 81.06%, while comparisons between *IL* and *L* presented a success rate of 76.51%. For both cases the *p-value* < 0.01. These results state the capability for interpolating vocal effort levels which entail lowering the voice tension.

On the other hand, *IH* samples compress those samples generated from interpolating vocal effort levels between high *H* and modal *M*. Comparisons between *IH* and *H* were successfully recognized 81.82% of the times with *p-value* < 0.01. However, when comparing *IH* with *M*, the success rate was slightly favorable with a 56.81% with *p-value* $= 0.0594$.

Results from table 5 and 6 demonstrate the capability of the proposed methodology to generate interpolated vocal effort levels with the following ordering: $L < IL < M \leq IH < H$.

4. Discussion

In previous work [10] a parametric model based on low order LPC was presented, however the model itself was sensitive to interpolation artefacts, which can lead to filter instabilities. So, the model presented serious difficulties for generating intermediate vocal effort levels. Other approaches are based on adding extra speech data in the corpus to cover the desired vocal effort levels to synthesize, but this creates a dependency between the model's flexibility and the corpus size. In this study we presented a methodology using parametric models based on ninth order polynomials, instead of the low order LPC model, not only for transferring vocal effort, but also for generating new interpolated vocal effort levels not present in the corpora recordings. The proposed methodology has been tested against previous work [10] in terms of vocal effort modification and synthesized signal quality. The results obtained from this comparison show that the presented methodology can reach the same degree of vocal effort modification as previous work while resulting in an improved signal quality in the final synthesis. The second test conducted has demonstrated that the presented method can be used for interpolating vocal effort levels. This has been possible due to linearity properties of the polynomial expressions used for interpolation. Despite presenting clear performance differences for most conversions, it is necessary to note the case when comparing *IH* against *M* where the effect is less robust. This could be a consequence of associating a wider vocal effort range to modal speech. The fact that statistical confidence increased for between *IH* and *H* makes us discard the possibility of the system to not being able to represent high vocal efforts. Thus, this uncertainty in intermediate high (*IH*) vocal effort with modal (*M*) levels could also be caused by non linear behavior of vocal effort perception or production.

Table 6: *Ordering of the synthesized vocal effort (VE) levels. The null hypothesis (H_0) was considered that users couldn't perceive any vocal effort level difference between each pair of samples.*

VE level pair	[%]	P-value
$H > IH$	81.8182	< 0.01
$IH > M$	56.8182	0.0594
$M > IL$	81.0606	< 0.01
$IL > L$	76.5152	< 0.01

These findings extend the previous conducted work [10] not only in overcoming the problem for generating interpolated vocal effort levels, but also achieving better performance in terms of signal quality.

In our proposed method, vocal effort models were adapted not only for phoneme identity but also for phoneme position in the recorded logatome. This decision was taken based on informal listenings of the corpora, which led us to realise that speakers realised a speaking pattern based on the syllable position within the utterance. Thus, this distinction was used in order to prevent effects due to the position of the syllable infer in the extraction of the harmonic spectral envelope models. In some logatomes presenting the same phonemes in several positions in the utterance, the achieved vocal effort modification varied from one position to the other. The fact of obtaining different harmonic spectral envelope models which produced different vocal effort degrees depending on the syllable position could be related with attack, decay, sustain and release situations. Whether this position distinction enhances the procedure or degrades its performance has not been evaluated for this corpora. However, when applying the model to sentences with semantic meaning, it might be important to consider the position of the syllables in the whole sentence. Moreover, when applying the model to expressive corpora with multiple emotions, the vocal effort modifications to be carried out, could depend on environment conditions such as whether the phoneme is stressed or not, position inside a stressed word or using accent-groups information such as [16]. The vocal effort model could be improved adding these additional information into the code-books.

The proposed method could also be combined with prosodic modifications such as pitch, energy or speech rate articulation. The combination of these signal modifications could be used to carry out expressive synthesized speech conveying different emotions.

The parameter β was introduced into the system's workflow

as a result of noticing that the multiplicative factors ($\boldsymbol{\alpha}$) despite achieving vocal effort modifications towards he expected target, the modification itself seemed laking some gain. This can be the consequence of scaling the harmonic spectral envelopes to fit into the range $[1, 2]$ before applying the quotient to obtain $\boldsymbol{\alpha}$. The use of the parameter β allowed to adjust signal modification degree. Multiplication factor ($\boldsymbol{\alpha}$) values are compressed between the range $[0.5, 2]$. $\boldsymbol{\alpha}$ values between $(1, 2]$ increase the harmonic energy, whereas values from $[0.5, 1)$ decrease the harmonics energy. To increase the modification magnitude $\boldsymbol{\alpha}$ values were powered, thus increasing the magnitude of the difference between amplifying ($\boldsymbol{\alpha} \in (1, 2]$) and attenuation ($\boldsymbol{\alpha} \in [0.5, 1)$) values. The β values used in the experiments were heuristically chosen in order to make modification noticeable. Two values were chosen, one for each gender, and they were held constant for all the synthesized utterances. Some improvement should be done to have better control of the magnitude of the modifications applied by the multiplicative factors matrix ($\boldsymbol{\alpha}$).

The current version of the proposed method is speaker dependent, follow-up work should focus on applying the model to several speakers and attempt to learn the variations that the model experiment when the users vocal effort moves around the different vocal effort levels. Finding any common pattern among the different speakers could allow to generalize the model making it speaker independent, thus probably avoiding the requirement for extreme vocal effort recordings to be able to generate the interpolated target levels.

5. Conclusions

The current work has presented a method of combining a polynomial model for vocal effort modification with HNM which allows us to transfer vocal effort from templates available in a corpus, as well as to generate interpolated vocal effort levels not present in the original recordings. A corpora specially designed for vocal effort research has been used in the experiments allowing us to isolate vocal effort from other effects usually present in natural speech presenting similar vocal effort conditions such as pitch or speed rate variations. The results present compelling evidence of the proposed system performing better than previous proposal [10]. Moreover, the results of a second test statistically support the proposed system's capability for generating interpolated vocal effort levels. Further work will focus on learning the variations that experiment the model when moving among different vocal effort levels. This knowledge could allow to generate a speaker independent model which would allow to carry out vocal effort modifications to any speaker without any previous information about their parameter's behavior in terms of vocal effort level shifting. This could be a crucial feature in applications where the TTS system has to perform several speaker registers such as in story-telling applications. Taking advantage of being a parametric model, it could potentially be employed by hidden Markov model (HMM) based speech synthesis systems in case appropriate acoustic features were being used.

6. Acknowledgments

We would like to thank Dr. Mark Schröder and DFKI for allowing us to use the NECA corpus which was specially designed for vocal effort research and Comisisionat per a Universitats i Recerca (CUR) from the DIUE of the Generalitat de Catalunya and the European Social Funds (2011 BE-DGR 01084) for funding to visit CSTR.

7. References

[1] A. Raux, B. Langner, A. W. Black, and M. Eskenazi, "Let's go: Improving spoken dialog systems for the elderly and non-natives," in *Eurospeech03*. Geneva, Switzerland: ISCA, September 2003, pp. 753–756.

[2] Y. Stylianou and O. Cappé, "A system for voice conversion based on probabilistic classification and a harmonic plus noise model," in *Proc. of the IEEE Int. conf. on acous., speech and signal processing*, vol. 1, 1998, pp. 281–284.

[3] C. Drioli, G. Tisato, P. Cosi, and F. Tesser, "Emotions and voice quality: experiments with sinusoidal modeling," in *VOQUAL03*, Geneva, Switzerland, August 2003, pp. 127–132.

[4] S.-J. Kim, J.-J. Kim, and M. Hahn, "Hmm-based korean speech synthesis system for hand-held devices," *IEEE Trans. on consumer electronics*, vol. 52, no. 4, pp. 1384 –1390, November 2006.

[5] C. Monzo, À. Calzada, I. Iriondo, and J. C. Socoró, "Expressive speech style transformation: voice quality and prosody modification using a harmonic plus noise model," in *Speech prosody 2010*, no. 100985, Chicago, May 2010.

[6] O. Türk and M. Schröder, "A comparison of voice conversion methods for transforming voice quality in emotional speech synthesis," in *Interspeech*. Brisbane, Australia: ISCA, September 2008, pp. 2282–2285.

[7] C. Gobl, E. Bennett, and A. N. Chasaide, "Expressive synthesis: how crucial is voice quality?" in *Proc. of IEEE workshop on speech synthesis*, 2002, pp. 91–94.

[8] E. Rank and H. Pirker, "Generating emotional speech with a concatenative synthesizer," in *5th Int. conf. on spoken language processing*, Sydney, Australia, 1998, pp. 671–674.

[9] M. Schröder and M. Grice, "Expressing vocal effort in concatenative synthesis," in *15th Int. conf. of phonetic sciences*, 2003, pp. 2589–2592.

[10] À. Calzada and J. C. Socoró, "Vocal effort modification through harmonics plus noise model representation," in *Proc. of the 5th int. conf. on advances in nonlinear speech processing*, ser. Lecture Notes in Computer Science, C. M. Travieso-González and J. B. Alonso-Hernández, Eds. Springer Berlin Heidelberg, 2011, vol. 7015, pp. 96–103.

[11] Y. Stylianou, "Harmonic plus noise models for speech, combined with statistical methods, for speech and speaker modification," Ph.D. dissertation, Ecole Nationale Superieure des Telecommunications, 1996.

[12] D. Erro, "Intra-lingual and cross-lingual voice conversion using harmonic plus stochastic models," Ph.D. dissertation, UPC, June 2008.

[13] R. J. McAulay and T. F. Quatieri, "Speech analysis/synthesis based on a sinusoidal representation," *IEEE Trans. on acous., speech and signal processing*, vol. 34, no. 4, pp. 744–754, August 1986.

[14] Sampa computer readable phonetic alphabet. [Online]. Available: www.phon.ucl.ac.uk/home/sampa

[15] S. Planet, I. Iriondo, E. Martínez, and J. A. Montero, "True: an online testing platform for multimedia evaluation," in *Proc. of the 2nd Int. workshop on emotion: Corpora for research on emotion and affect at the 6th conf. on lang. resources & evaluation (LREC 2008)*, Marrakech, 2008.

[16] D. Erro, I. Sainz, I. Luengo, I. Odriozola, J. Sánchez, I. Saratxaga, E. Navas, and I. Hernáez, "Hmm-based speech synthesis in basque language using hts from ahotts to aho-hts," in *FALA 2010*, no. 978-84-8158-510-0, Vigo, 2010, pp. 67–70.

Vietnamese HMM-based Speech Synthesis with prosody information

Anh-Tuan DINH[2],Thanh-Son PHAN[1], Tat-Thang VU[2], Chi-Mai LUONG[2]

[1]Faculty of Information Technology, Le Qui Don Technical University, Hanoi, Vietnam
[2]Institute of Information Technology, Vietnam Academy of Science and Technology, Hanoi, Vietnam

[1]sonphan.hts@gmail.com, [2]{anhtuan,vtthang,lcmai}@ioit.ac.vn

Abstract

Generating natural-sounding synthetic voice is an aim of all text to speech system. To meet the goal, many prosody features have been used in full-context labels of an HMM-based Vietnamese synthesizer. In the prosody specification, POS and Intonation information are considered not as important as positional information. The paper investigates the impact of POS and Intonation tagging on the naturalness of HMM-based voice. It was discovered that, the POS and Intonation tags help reconstruct the duration and emotion in synthesized voice.

Index Terms: Vietnamese speech synthesis, tone characteristics, tonal language, prosody tagging, part-of-speech, hidden Markov models

1. Introduction

In HMM-based speech synthesis systems, the prominent attribute is the ability to generate speech with arbitrary speaker's voice characteristics and various speaking styles without large amount of speech data1].

The naturalness of a Vietnamese TTS system is mainly affected by prosody. Prosody consists of accent, intonation and Vietnamese tones (6 tones). The features with part-of-speech (POS) tagging have close relationships and determine the naturalness and the intelligibility of synthesized voice.

Vietnamese tones consist of level, falling, broken, curve, rising, drop. One syllable can change its meaning when it goes with different tones. So the tonal feature has a strong impact on the intelligibility of synthetic voice. The following Table 1 shows an example about the name, the mark of tone in Vietnamese.

Table 1. *Six tones in Vietnamese*

Name	Tone mark	Example
LEVEL (ngang)	Unmarked	ta – me
FALLING (huyền)	Grave	tà – bad
BROKEN (ngã)	Tilde	tã - napkin
CURVE (hỏi)	Hook above	tả - describe
RISING (sắc)	Acute	tá – dozen
DROP (nặng)	Dot below	tạ - quintal

2. Text to Speech System

A TTS system, showed in Figure 1, is the production of speech from text. It includes the following stages[5]:

- Text tokenization splits the input text stream into smaller units named sentences and tokens. In the phase, written forms of Vietnamese syllables are discovered and tagged. The process is called tokenization.
- Text normalization decodes non-standard word into one or more pronounceable words. Non-standard tokens including numbers, dates, time, abbreviations… are normalized in the phase. The process is also called homograph disambiguation.
- Text parsing investigates lexical, syntactic structures from words which are used for pronunciation and prosodic modeling stages. The stage consists of POS tagging and chunking.

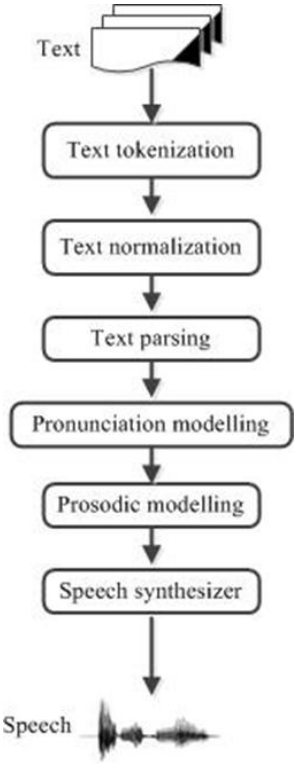


Figure 1: *A TTS system.*

- Pronunciation modeling maps each word to its phonemes. It looks up the words in a lexicon or use grapheme to phoneme (G2P) rules to finish the task. Accents and tones are assigned
- Prosodic modeling predicts the prosody of sentences. Sentence-level stress is identified, the intonation is assigned to sentences which make melody or tune of entire sentences.
- Speech synthesizer generates the speech waveforms from the above information. In Hidden Markov Model-based synthesis, a source filter paradigm is used to model the speech acoustics; information from previous stages are used to make full-context label file of each phoneme in the input sentence. Excitation (fundamental

frequency F_0), spectrum and duration parameters are estimated from recorded speech and modeled by context-dependent HMMs.

Figure 2 is an example of full context model used in HMM-based Speech Synthesis:

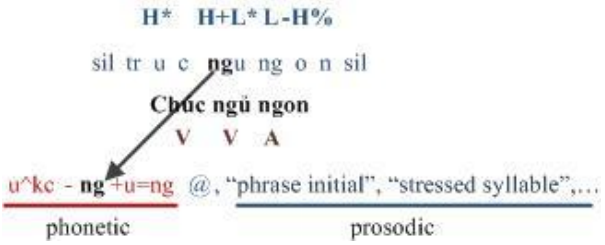


Figure 2: *A HMM full context model.*

Based on the full context model, HMM-based TTS is very flexible easy to add more prosodic information.

3. The improvement of tonal and accentual features

It is thought that tone lies on vowel; however, tone plays an important role on all over a syllable in Vietnamese. However, tonal features are not as explicit as other features in speech signal. According to Doan Thien Thuat[10], a syllable's structure can be described in Table 2:

Table 2. *Structure of Vietnamese syllable.*

Tone			
[Initial]	Final		
	[Onset]	Nucleus	[Coda]

In the first consonant, we can hear a little of the tone. Tone becomes clearer in rhyme and finished completely at the end of the syllable. The pervading phenomenon determines the non-linear nature of tone. So, with mono syllabic language like Vietnamese, a syllable can't easily be separated into small acoustic parts like European languages.

In syllable tagging process, contextual features must be considered. There are many contextual factors (ex, phonetic, stress, dialect, tone) affecting the signal spectral, fundamental frequency and duration. In additional, constructing a decision tree to classify the phonemes based on contextual information. The construction of the decision is very important in HMM-based Vietnamese TTS system[6].

Some contextual information include tone, accent, part-of-speech, was considered as follows[9]:

- Phoneme level:
  - Two preceding, current, two succeeding phonemes
  - Position in current syllable (forward, backward)
- Syllable level:
  - Tone types of two preceding, current, two succeeding syllables
  - Number of phonemes in preceding, current, succeeding syllables
  - Position in current word (forward, backward)
  - Stress-level
  - Distance to {previous, succeeding} stressed syllable
- Word level:
  - Part-of-speech of {preceding, current, succeeding} words
  - Number of syllables in {preceding, current, succeeding} words
  - Position in current phrase
  - Number of content words in current phrase {before, after} current word
  - Distance to {previous, succeeding} content words
  - Interrogative flag for the word
- Phrase level:
  - Number of {syllables, words} in {preceding, current, succeeding} phrases
  - Position of current phrase in utterance
- Utterance level:
  - Number of {syllables, words, phrases} in the utterance

4. Intonation in Vietnamese

In order to present intonation, we use Tones and Break Indices (ToBI) in intonation transcription phase. ToBI is a framework for developing a widely accepted convention for transcribing the intonation and prosodic structure of spoken sentences in various languages. ToBI framework system is supported in HTS engine. The primitives in a ToBI framework system are two tones, low (L) and high (H). The distinction between the tones is paradigmatic. That is L is lower than H in the same context. Utterances can consist of one or more intonation phrases. The melody of an intonation phrase is separated into a sequence of elements, each made up of either one or two tones. In our works, the elements can be classified into 2 main classes[3]:

4.1. Phrase-final intonation

Intonation tones, mainly phrase-final tones, were analyzed in our work. Boundary tones are associated with the right edge of the prosodic phrase and mark the end of a phrase. It can be established in Vietnamese that, a high boundary tone can change a declarative into an interrogative. To present the boundary tone, 'L-L%', 'L-H%' tags are used. 'L-L%' refers to a low tone; and 'L-H%' describes a high tone. This is a common declarative phrase. The 'L-L%' boundary tone causes the intonation to be low at the end of the prosodic phrase. On the other hand, the effect of 'L-H%' is that first it will drop to a low value and then it will rise towards the end of the prosodic phrase.

4.2. Pitch Accent

Pitch Accent is the falling or rising trends in the top line or baseline of pitch contour. Most noun, verb and adjective in Vietnamese are accented words. An 'H*' (high-asterisk) tends to produce a pitch peak while an 'L*' (low-asterisk) pitch accent produces a pitch trough. In addition, the two other tag 'L+H*' and 'H+L*' are also used. 'L+H*' rises steeply from a much lower preceding F_0 value while 'H+L*' falls from a much higher preceding F_0 value.

It was showed in the experiment that: the intonation tags add valuable contextual information to Vietnamese syllables in training process. Spoken sentences can be distinguished easily between declarative and interrogative utterances. Import information in a speech is strongly highlighted.

5. Part of Speech

A POS tag is a linguistic category assigned to a word in a sentence based upon its morphological behavior. Words are classified into POS categories such as noun (N), verb (V), adjective (A), pronoun (P), determine (D), adverb (R), apposition (S), conjunction (C), numeral (M), interjection (I) and residual (X). Words can be ambiguous in their POS categories. The ambiguity normally solved by looking at the context of the word in the sentence.

Automatic POS tagging is processed with Conditional Random Fields. The training of CRFs model is basically to maximize the likelihood between model distribution and empirical distribution. So, CRFs model training is to find the maximum of a log - likelihood function.

Suppose that training data consists of a set of N pairs, each pair includes an observation sequence and a status sequence, D = {(x(i),y(i))} $\forall i = 1 \ldots N$. Log-likelihood function:

$$l(\theta) = \sum_{x,y} \tilde{p}(x,y) \log(p(y \mid x, \theta)), \quad (1)$$

Here, $\theta(\lambda_1, \lambda_2, ..., \mu_1, \mu_2, ,...)$ is the parameter of the model and $\tilde{p}(x,y)$ is concurrent empirical experiment of x, y in training set. Replace $p(y \mid x)$ of (1), we have:

$$l(\theta) = \sum_{x,y} \tilde{p}(x,y) \left[\sum_{i=1}^{n+1} \lambda f + \sum_{i=1}^{n} \mu g \right] - \sum_{x} \tilde{p}(x) \log Z, \quad (2)$$

Here, $\lambda(\lambda_1, \lambda_2, ..., \lambda_n)$ and $\mu(\mu_1, \mu_2, ..., \mu_m)$ are parameter vectors of the model, f is a vector of transition attributes, and g is a vector of status attributes.

6. Experiment and Evaluation

In the experiment, we used phonetically balanced 400 in 510 sentences (recorded female and male voices, Northern dialect) from Vietnamese speech database for training. All sentences were segmented at the phonetic level. The phonetic labeling procedure was performed as text-to-phoneme conversion through a forced alignment using a Vietnamese speech recognition engine[11]. During the text processing, the short pause model indicates punctuation marks and the silence model indicates the beginning and the end of the input text.

For the evaluation, we used remain 110 sentences in the speech database, these sentences are used as synthesize data for testing and evaluating. Feature vector consists of spectral, tone, duration and pitch parameter vectors: spectral parameter vector consists of 39 Mel-frequency cepstral coefficients including the zero-th coefficient, their delta and delta-delta coefficients; pitch feature vector consists of logF_0, its delta and delta-delta[7].

A couple of comparisons of synthesized speech qualities, include male and female speech models with only tone and with additional POS, stress and intonation. The information is added to full context model of each phoneme in a semi automatic way.

6.1. Objective test

The objective measurement is described through comparing of pitch contour between natural speech and synthesized testing sentences in both cases and showed in Figure 3 and Figure 4.

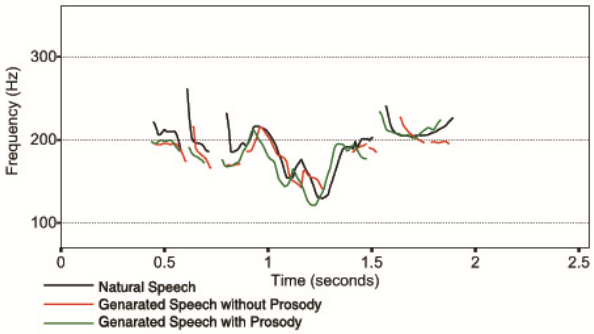


Figure 3: *ComparisonF_0 contour extracted from utterance "Anh có cái gì rẻ hơn không?"of Natural Speech and Generated Speeches, male voice*

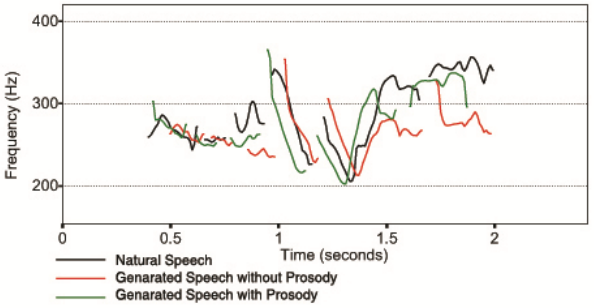


Figure 4: *ComparisonF_0 contour extracted from utterance "Anh có cái gì rẻ hơn không?"of Natural Speech and Generated Speeches, female voice*

6.2. MOS test

As a further subjective evaluation, MOS tests were used to measure the quality of synthesized speech signals in comparison with natural ones. The rated levels were: bad (1), poor (2), fair (3), good (4), and excellent (5). In this test, a hundred sentences were randomly selected. With three types of audio, (1) natural speech signals, (2) the synthetic speech signals without POS, accent and intonation, and (3) the synthetic speech signals with POS, accent and intonation, the number of listeners were 50 people. The speech signals were played in random order in the tests.

Table 3 shows the mean of opinion scores which were given by all the subjects. The MOS result implied that the quality of natural speech is from good to excellence, and the quality of synthesis speech is from fair to good.

Table 3. *Resultsof the MOS test*

Speech	Mean Opinion Score
Natural	**4.53**
Without POS, Accent, Intonation	**3.15**
With POS, Accent, Intonation	**3.89**

7. Conclusions

The experimental results, shown that POS and prosody information do contribute to the naturalness (specifically in terms of pitch) of a TTS voice when it forms part of a small phoneme identity-based feature set in the full context HTS labels. The experiments were limited by Northern dialect

corpus. It would be prudent to test the effects on the other dialects, especially the South Central dialect.

The proposed tonal features can improve the tone intelligibility for generated speech. In addition, beside tonal features, the proposed POS and prosody features give the better improvement of the synthesized speech quality. These results confirm that the tone correctness and prosody of the synthesized speech is significantly improved and more naturalness when using most of the extracted speech features. Future work includes the improvement of text processing automatically and work on the contextual information.

8. Acknowledgements

This work was partially supported by ICT National Project KC.01.03/11-15 "Development of Vietnamese - English and English - Vietnamese Speech Translation on specific domain". Authors would like to thank all staff members of Department of Pattern Recognition and Knowledge Engineering, Institute of Information Technology (IOIT) - Vietnam Academy of Science and Technology (VAST) for their support to complete this work.

9. References

[1] J.Yamagishi, K.Ogata, Y.Nakano, J.Isogai, T.Kobayashi,"HSMM-Based Model adaptation algorithms for Average-Voice-Based speech synthesis", 77–80, ICASSP 2006.

[2] K. Tokuda, T. Yoshimura, T. Masuko, T. Kobayashi, and T. Kitamura,"Speech parameter generation algorithms for HMM-based speech synthesis", 1315–1318, Proc. ICASSP,June 2000.

[3] H. Mixdorff, H. B. Nguyen, H. Fujisaki, C. M. Luong,"Quantitative Analysis and Synthesis of Syllabic Tones in Vietnamese", 177-180, Proc. EUROSPEECH, Geneva, 2003.

[4] Phu Ngoc Le, Eliathamby Ambikairajah, Eric H.C. Choi,"Improvement of Vietnamese Tone Classification using FM and MFCC Features", 01-04, Computing and Communication Technologies RIVF 2009.

[5]Schlunz, GI, Barnard, E and Van Huyssteen, GB,"Part-of-speech effects on text-to-speech synthesis", 257-262, 21st Annual Symposium of the Pattern Recognition Association of South Africa (PRASA), Stellenbosch, South Africa, 22-23 November 2010.

[6] Son Thanh Phan, Thang Tat Vu, Cuong Tu Duong, and Mai Chi Luong,"A study in Vietnamese statistical parametric speech synthesis base on HMM", 01-06, IJACST, Vol. 2, No. 1, Jan 2013.

[7] Son Thanh Phan, Thang Tat Vu, and Mai Chi Luong,"Extracting MFCC, F_0 feature in Vietnamese HMM-based speech synthesis, International Journal of Electronics and Computer Science Engineering", 2(1):46-52, Jan 2013.

[8]Tang-Ho Le, Anh-Viet Nguyen, Hao Vinh Truong, Hien Van Bui, and Dung Le,"A Study on Vietnamese Prosody", 63-73, New Challenges for Intelligent Information and Database SystemsStudies in Computational Intelligence Volume 351, 2011.

[9] Thang Tat Vu, Mai Chi Luong, Satoshi Nakamura,"An HMM-based Vietnamese Speech Synthesis System", 116 - 121, Proc. Oriental COCOSDA, 2009.

[10]T.T. Doan, "Vietnamese Acoustic", Vietnamese National Editions, Second edition, 2003.

[11]T.T Vu, D.T. Nguyen, M.C. Luong, J.P Hosom,"Vietnamese large vocabulary continuous speech recognition", 1689-1692, Proc. INTERSPEECH, 2005.

Context labels based on "bunsetsu" for HMM-based speech synthesis of Japanese

Hiroya Hashimoto[1], *Keikichi Hirose*[1], *Nobuaki Minematsu*[2]

[1]Department of Information and Communication Engineering
[2]Department of Electrical Engineering and Information Systems,
the University of Tokyo, Japan
{hiroya, hirose, mine}@gavo.t.u-tokyo.ac.jp

Abstract

A new set of context labels was developed for HMM-based speech synthesis of Japanese. The conventional labels include those directly related to sentence length, such as number of "mora" and order of breath group in a sentence. When reading a sentence, it is unlikely that we count its total length before utterance. Also a set of increased number of labels is required to handle sentences with various lengths, resulting in a less efficient clustering process. Furthermore, labels related to prosody are mostly designed based on the unit "accent phrase," whose definition is somewhat unclear; it is not uniquely defined for a given sentence, but also is affected by other factors such as speaker identity, speaking rate, and utterance style. Accent phrase boundaries may be labeled differently for utterances of the same content, and this situation affects other labels, because of numerical labeling scheme counted from the sentence/breath-group initial. In the proposed labels, "bunsetsu" is used instead. Also, we only view its relations with preceding and following "bunsetsu's." Thus labels not related to the sentence lengths are obtained, with easier automatic prediction only from sentence representations. Validity of the proposed labels was shown through speech synthesis experiments.

Index Terms: speech synthesis, context labels, linguistic information

1. Introduction

Recently, statistical framework, such as hidden Markov modeling, has been successfully introduced to analysis-synthesis-based speech synthesis systems[1]. Although there still are some degradations in speech quality as compared to waveform concatenation methods, HMM-based speech synthesis is now widely used, since it can generate speech in various voice qualities and speaking styles from a very limited speech corpus through adaptation/conversion techniques[2, 3]. Although HMM's are commonly used in speech recognition, they are differently organized in speech synthesis. In the case of speech recognition, since the aim is to recognize phonemes, one HMM is trained for each phoneme separately for surrounding phonemes. Other factors affecting phoneme features, such as positions in an utterance, accent types, etc. are not counted. However, in the case of speech synthesis, variations of phoneme features need to be realized as correctly as possible. Therefore, various contextual factors are taken into account, and a number of context labels are prepared to represent these factors. Since combinations of these labels are huge, we faced to the problem of data sparseness, if we try to train an HMM for each combination. Grouping of conditions are commonly done before HMM training to solve the problem. Also several methods have been developed to reduce context numbers, including one to select important labels by finding relation of labels using Bayesian networks[4], and one to use F0 digitized for each phoneme instead of accent types as prosody contexts[5]. Resulting synthetic speech, however, includes some unnaturalness. One possible reason for this situation resides in the design of the context labels.

The context labels widely used for Japanese speech synthesis are those used in HTS[6], a well-known HMM-based speech synthesizer. They include following two problems. First, labels related to prosody are designed based on "accent phrase," whose definition has an ambiguity and cannot be decided only from text. It may be subject to change by utterance speeds and styles. The second problem is the sequential numbering from the sentence/breath-group initial adopted in some labels. In order to cope with sentences with arbitrary lengths, a large number of labels are required. Moreover, labels can be differently assigned for the same phrases (with similar prosodic features), but in different sentences. This label ambiguity may also happen even for the same sentences, when they have different accent phrase boundaries.

In order to solve this situation, we newly developed a set of context labels based on "bunsetsu" instead of "accent phrase." "Bunsetsu" is defined as a basic unit of Japanese syntax and pronunciation consisting of content word(s) followed or not followed by particles. "Bunsetsu" boundaries can be predicted only from text with high performance. Furthermore, we avoided to use positions in sentence/breath-group.

The rest of the paper is organized as follows: Newly proposed context labels are presented and compared with conventional context labels in section 2. In section 3, the proposed context labels are evaluated through listening test of synthetic speech. Section 4 concludes the paper with some discussions.

2. Context labels

2.1. Context labels for HTS (conventional context labels)

Table 1 shows context labels adopted in HTS Japanese speech synthesizer. Labels related to prosody are designed based on "accent phrase," which is defined as an utterance unit with a pair of rise and fall of fundamental frequency (F0) contour (an accent component). Mora with F0 fall is crucial for perception of Japanese lexical accent and is called an accent nucleus. The context labels are designed assuming one accent component in each accent phrase, though an accent phrase can have a minor accent component, called secondary accent. Labeling of accent

Utterance 1: yamano/ueno/mukooni/kireina/hanaga...
1 2 3 4 5
Utterance 2: yamanoueno/mukooni/kireina/hanaga...
1 2 3 4
(Over the hill top, there are beautiful floors...)

Figure 1: An example of accent phrase labeling

phrases for speech corpus of HMM training usually conducted manually by labelers referring to texts and speech sounds. Certain inconsistencies are unavoidable in the labeling process. Especially, it is often difficult to tell an minor F0 movement being as "reduced" accent by de-focusing, secondary accent, or no accent component. Although there have been several attempts for automatic extraction of accent phrases, their performances are not high enough. To begin with, there are number of cases hard to exactly tell whether a (linguistic) phrase consists of one accent phrase or two (or more) accent phrases. The cases may increase when we handle spontaneous speech. When a sentence is uttered in a different speaking style or by a different speaker, the accent phrases may change, because they are units of "utterance." Regardless of these accent phrase ambiguities, accent phrases are predicted only from texts in HMM-based speech synthesis. This situation may not cause a serious problem when handling a speech corpus carefully (and thus consistently) uttered in reading style by a professional speaker. Because of the cost of labeling, however the same accent labeling is often used for a new speech corpus by a different speaker or in a different speaking style. This may increase errors in accent phrase labeling.

Since accent phrases are sequentially numbered in a breath group, a difference in accent phrase labeling may spread to other parts as shown in Fig 1. Symbol "/" indicates accent phrase boundary. Context label "position of the current phrase in the current breath group" (in Table 1) is totally differently labeled, though the difference between two utterances is the existence of accent phrase boundary after "yamano" in utterance 1. Also since the context labels include those on lengths of sentence, breath group, and accent phrase, which are counted by number of morae, unnecessarily large numbers need to be prepared to cope with various sentences and speaking styles. Although these labels are usually discarded (or summarized) through context clustering, some of these labels sometimes degrade synthetic speech quality.

2.2. Proposed context labels

In order to solve the problems listed in the previous section, a new set of context labels is constructed as shown in Table 2. Two labels ID1 and ID2 are prepared to represent POS (part-of-speech) and S-POS (supplemental POS), respectively, based on the Unidic Japanese dictionary for morpheme analysis[7]. ID1 includes following parts-of-speech: verb, noun, adjective, adjectival verb, adnominal, adverb, pronoun, interjection, particle, auxiliary verb, prefix, suffix, sentence initial, short pause, and sentence end. The last three items are included, since pauses largely affect other prosodic features. ID2 is supplemental to ID1 and indicates the role of the word. It includes: "can be used as content word," "can be used as particle," general, common noun, numeral, proper noun, noun like, verb like, adjective like, adjectival verb like, nominative particle, "particle that attaches to a phrase and acts on the whole phrase," adverbial particle, conjunctive particle, binding particle, sentence-end particle, stem of auxiliary verb, "tari" conjugation, and filler.

Table 1: Context labels adopted in Japanese HTS

Previous phoneme identity
Current phoneme identity
Next phoneme identity
Position of the current mora in the current accent phrase
Difference between accent type and position of the current mora
POS of the previous word
Inflected form of the previous word
Conjugation type of the previous word
POS of the current word
Inflected form of the current word
Conjugation type of the current word
POS of the next word
Inflected form of the next word
Conjugation type of the next word
Number of morae of the previous accent phrase
Accent type of the previous accent phrase
Connection intensity between the previous accent phrase and the current accent phrase
Pause existence between the previous accent phrase and the current accent phrase
Number of morae in the current accent phrase
Accent type in the current accent phrase
Connection intensity between the previous accent phrase and the next accent phrase
Position of the current accent phrase in the current breath group
Interrogative sentence or not
Number of morae of the next accent phrase
Accent type of the next accent phrase
Connection intensity between the next accent phrase and the current accent phrase
Pause existence between the next accent phrase and the current accent phrase
Number of morae of the previous breath group
Number of morae of the current breath group
Position of the current breath group in the sentence
Number of morae of the next breath group
Number of morae of the sentence

The new labels have following three major differences from those of HTS.

i) "Bunsetsu" is used instead of "accent phrase." Since "bunsetsu" is a grammatically defined unit, it can be identified uniquely from text. Also "very long" samples found in accent phrases do not occur, and maximum number can be set small for "bunsetsu" length counted in mora unit.

ii) High (1) or Low (0) is assigned to each moraic F0 pattern instead of accent types. Japanese word accent types are often stylized with high and low patterns of F0 contours in mora unit. High-low assignment can be automatically done for each mora when accent types are given. Due to accent concatenation, Japanese word accent in continuous speech may change from that of isolated utterance. When two content words concatenates, they are uttered together in one accent type. However, when we emphasize one of two words, concatenation may not happen; two words are uttered with their original accent types. If

Table 2: Context labels of the proposed method

Previous phoneme identity
Current phoneme identity
Next phoneme identity
F0 level of the previous mora (0:Low, 1:High)
F0 level of the current mora (0:Low, 1:High)
F0 level of the next mora (0:Low, 1:High)
Position of the current mora in the current word (counted from word initial)
Position of the current mora in the current word (counted from word end)
Position of the current mora in the current "bunsetsu" (counted from "bunsetsu" initial)
Position of the current mora in the current "bunsetsu" (counted from "bunsetsu" end)
Number of morae of the previous word
Number of morae of the current word
Number of morae of the next word
Number of morae of the previous "bunsetsu"
Number of morae of the current "bunsetsu"
Number of morae of the next "bunsetsu"
POS ID1 of the previous word
POS ID1 of the current word
POS ID1 of the next word
POS ID1 of the content word in the previous "bunsetsu"
POS ID1 of the content word in the current "bunsetsu"
POS ID1 of the content word in the next "bunsetsu"
S-POS ID2 of the previous word
S-POS ID2 of the current word
S-POS ID2 of the next word
S-POS ID2 of the content word in the previous "bunsetsu"
S-POS ID2 of the content word in the current "bunsetsu"
S-POS ID2 of the content word in the next "bunsetsu"
Whether the current mora consisting of only one short vowel or not
Whether the current mora containing long vowel or not

we use "accent type," these phenomena are explained in complex, but they can be simplified with high-low pattern representations. Secondary accents can also be labeled easily.

iii) Only relative positions are adopted. Absolute positions, such as position of breath group in sentence, and position of accent phrase in breath group, are not used. Total lengths of sentence and breath group are not used. These can reduce the total number of labels and can prevent labeling ambiguities affecting other parts.

There are minor differences in the labeling: a label identifying long vowel from singleton is included, and a label identifying interrogative sentence from declarative sentence is deleted. The second change is done, because no interrogative sentences are included in the corpus used in the experiment. As for the first one, we can also include long vowels in the phoneme set instead. (In HTS, a long vowel is represented by two short vowels. This sometimes causes confusions with two short vowels, which are not merged to a long vowel.)

Since high/low F0 level of a mora is tightly related to those of preceding and following morae, in the context clustering process, combinations of labels of "F0 levels of previous, current, and next morae" are included in the question sets. For instance, "previous (0) + current (1) + next (0)" is included.

3. Speech synthesis experiment

HMM-based speech synthesis is conducted for two cases, using HTS context labels and using proposed labels. Synthetic speech from the two cases is compared in their naturalness through a listening test.

3.1. Method

From ATR continuous speech corpus B set[8], utterances by male speaker (MMI) and female speaker (FTY) are selected for the synthesis experiment. (Speech syntheses for speaker MMI and speaker FTY are conducted.) Each speaker uttered 503 sentences, and 450 sentences are used for HMM training, with rest 53 sentences for testing. Speech signals are sampled at 16 kHz sampling rate, and STRAIGHT analysis[9] is used to extract spectral envelope, F0, and aperiodicity with 5-ms frame shift. Minimum and maximum values for F0 extraction are set to 120 Hz and 400 Hz for the female speaker, and 60 Hz and 250 Hz for the male speaker. The spectral envelope is converted to mel-cepstral coefficients using a recursion formula. The feature vector is 138 dimensional, consisting of 40 mel-cepstral coefficients including the 0th coefficient, the logarithm of fundamental frequency, 5 band-aperiodicity (0–1 kHz, 1–2 kHz, 2–4 kHz, 4–6 kHz, 6–8 kHz) and their delta and delta-delta coefficients. HMM with five-state left-to-right model topology is used. Output from each state is represented by a single Gaussian with diagonal covariance matrix. Context clustering is conducted using binary decision trees with MDL stop criterion. HMM training is conducted by HTS-2.1.

Context labels related to lexical accents are manually labeled, while those of part-of-speech are automatically labeled using open source Japanese parser "mecab"[10] with manual correction.

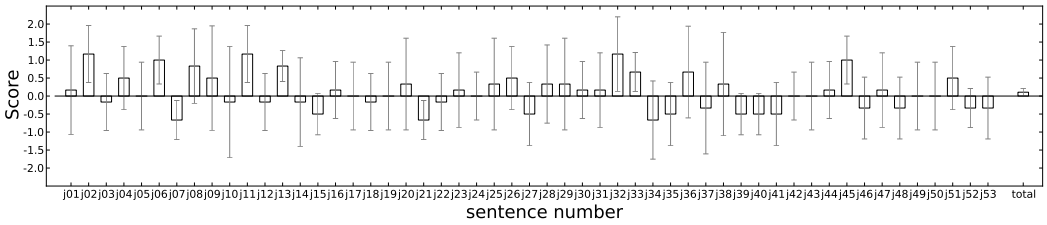
6 native speakers of Japanese evaluated the synthetic speech. They are asked to listen pairs of synthetic speech (one is by conventional labels and the other is by proposed labels), and select one on 5-scale scoring (2: one by proposed labels is clearly better than one by conventional labels, 1: one by proposed labels is better than one by conventional labels, 0: same, -1: one by conventional labels is better than one by proposed labels, -2: one by conventional labels is clearly better than one by proposed labels).

3.2. Result

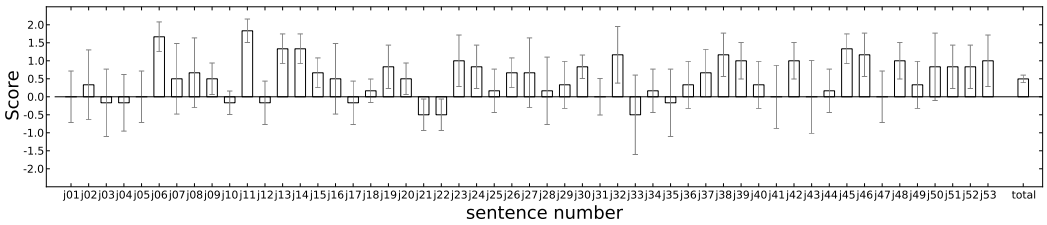
Results are shown in Fig. 2 with a confidence interval of 95%. The average score over the 53 test sentences is 0.109 with 0.106 confidence interval in significance level of 5% for FTY. and 0.497 with 0.105 for MMI. From the results, it is found that the proposed method improves the quality of the synthetic speech.

Fig. 3 compares generated F0 contours by the two sets of context labels. Some unnatural movements in the F0 contour by conventional context labels are settled by proposed context labels, indicating that the lexical accents can be well represented only by high-low F0 labeling of each mora. One of the major reasons of the degradation by the proposed labeling is the duration control. Inspection of decision trees for durations indicates that context labels on "number of morae" and on "position of mora" often appear near the top nodes for the proposed context labels, while they do not appear for the conventional context labels. Further studies on the labels are necessary from this viewpoint.

Combinations of "F0 levels of previous, current, and next morae" appear near the top node of the decision trees for funda-

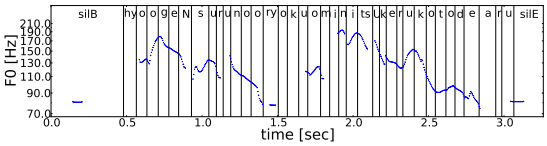


(a) FTY (female speaker)

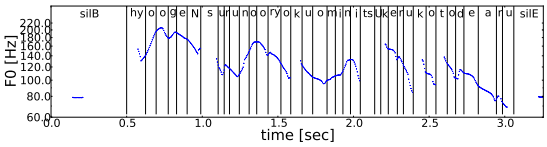


(b) MMI (male speaker)

Figure 2: Result of subjective test



(a) conventional



(b) proposed

Figure 3: Comparison of F0 contours for a Japanese sentence: hyoogeNsurunooryokuo minitsukerukotodearu (Is is to obtain an ability of expressing.). From top to bottom, F0 contour generated with conventional context labels, that generated with the proposed context labels. (Speaker MMI)

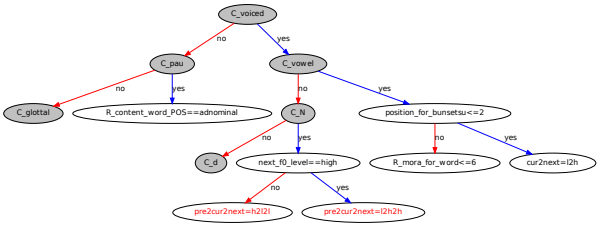


Figure 4: An example of decision tree for F0 (Speaker MMI, 2nd state). The nodes in red are combinations of "F0 levels of previous, current, and next morae."

mental frequencies as shown in Fig 4. This result indicates the correlations of labels. Further experiments are necessary to find the best combinations.

4. Conclusion and disucussion

A new set of context labels was constructed for HMM-based speech synthesis. Since the new labels are not using absolute positions of units in utterances, efficient and compact labeling is possible for speech corpus with various lengths. The labels also adopts "bunsetsu" instead of "accent phrase," enabling consistent labeling only from text. These features facilitate the HMM training process, and thus improve the synthetic speech quality, which is proved through a listening test of synthetic speech. The effect of the new labels may come clearer when handling long sentences, which are not included in the current speech corpus. (Maximum length of the sentence included in the current speech corpus is around 60 morae.)

Although, in the current experiment, the new labels and those of HTS are assigned manually, automatic labeling will be easier for the new labels. We already have conducted a preliminary speech synthesis experiment using speech corpus with automatically assigned labels, and confirmed that no apparent degradation observable for the new labels. We are now further improving the labels so that they can well handle various styles of speech.

The labeling scheme should be also beneficial to languages other than Japanese: for instance, in English HTS, the context labels include ones such as "position of the current syllable in the current word," "position of the current syllable in the current phrase," and "number of syllables in the utterance." These labels may cause problems similar to Japanese.

5. References

[1] T. Yoshimura, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, "Speaker interpolation in HMM-based speech synthesis system," Proc. EUROSPEECH, pp. 2523–2526, 1997.

[2] J. Yamagishi, K. Onishi, T. Masuko, and T. Kobayashi, "Acoustic modeling of speaking styles and emotional expressions in HMM-based speech synthesis," IEICE Trans. Inf. & Syst., vol. E88-D, no. 3, pp. 503–509, 2005.

[3] T. Nose, Y. Kato, and T. Kobayashi, "A speaker adaptation technique for MRHSMM-based style control of synthetic speech," Proc. ICASSP, pp. 833–836, 2007.

[4] Heng Lu, and Simon King, "Bayesian Networks to find relevant context features for HMM-based speech synthesis," Proc. INTERSPEECH, 2012.

[5] T. Nose, K. Ooki, T. Kobayashi, "HMM-based speech synthesis with unsupervised labeling of accentual context based on F0 quantization and average voice model," Proc. ICASSP, pp. 4622–4625, 2010.

[6] HTS, http://hts.sp.nitech.ac.jp/

[7] Unidic, http://sourceforge.jp/projects/unidic/

[8] A. Kurematsu, Kazuya Takeda, Yoshinori Sagisaka, Shigeru Katagiri, Hisao Kuwabara, and Kiyohiro Shikano, "ATR Japanese speech database as a tool of speech recognition and synthesis," Speech Communication, vol. 9, pp. 357–363, 1990.

[9] H. Kawahara, I. Matsuda-Katsuse, and A. de Cheveigné, "Restructuring speech representations using a pitch-adaptive time-frequency smoothing and an instantaneous-frequency-based F0 extraction: possible role of a repetitive structure in sounds," Speech Communication, vol. 27, no. 3–4, pp. 187–207, 1999.

[10] Mecab, http://mecab.googlecode.com/svn/trunk/mecab/doc/index.html

Using Adaptation to Improve Speech Transcription Alignment in Noisy and Reverberant Environments

Y. Mamiya[1], *A. Stan*[2], *J. Yamagishi*[1,3], *P. Bell*[1], *O. Watts*[1], *R.A.J. Clark*[1], *S. King*[1]

[1]Centre for Speech Technology Research, University of Edinburgh, United Kingdom
[2]Department of Communications, Technical University of Cluj-Napoca, Romania
[3]National Institute of Informatics, Tokyo, Japan

{yoshitaka.mamiya,jyamagis,owatts}@inf.ed.ac.uk,robert@cstr.ed.ac.uk
adriana.stan@com.utcluj.ro, {Peter.Bell,Simon.King}@ed.ac.uk

Abstract

When using data retrieved from the internet to create new speech databases, the recording conditions can often be highly variable within and between sessions. This variance influences the overall performance of any automatic speech and text alignment techniques used to process this data. In this paper we discuss the use of speaker adaptation methods to address this issue. Starting from a baseline system for automatic sentence-level segmentation and speech and text alignment based on GMMs and grapheme HMMs, respectively, we employ Maximum A Posteriori (MAP) and Constrained Maximum Likelihood Linear Regression (CMLLR) techniques to model the variation in the data in order to increase the amount of confidently aligned speech. We tested 29 different scenarios, which include reverberation, 8 talker babble noise and white noise, each in various combinations and SNRs. Results show that the MAP-based segmentation's performance is very much influenced by the noise type, as well as the presence or absence of reverberation. On the other hand, the CMLLR adaptation of the acoustic models gives an average 20% increase in the aligned data percentage for the majority of the studied scenarios.

Index Terms: speech alignment, speech segmentation, adaptive training, CMLLR, MAP, VAD

1. Introduction

For any corpus-based speech synthesis system or automatic speech recognition system, one of the most important considerations is the selection of high quality speech data for training purposes. For a limited number of languages, such as English, Spanish, French, and German, developers can chose from many widely available specifically-prepared resources. However, for most of the world's languages such speech databases are not readily available. Even for apparently well-resourced languages, the specific content of available data might not be suitable for a particular need – for example data in a sports news speaking style would probably not be as readily available as broadcast news data. In such situations, new resources either need to be recorded from scratch, or created from existing sources such as podcasts or audiobooks. To manually process sufficient data – for example, transcribing the words – would be time consuming and expensive, and thus a barrier to creating speech recognition or synthesis systems for new domains or new languages.

Automatic alignment of speech with imperfect transcripts has already been well addressed in the previous work of others, for example [1, 2, 3, 4, 5, 6, 7]. Unfortunately, all of these approaches make use of expert knowledge and/or expensive resources, such as very good speaker-independent acoustic models or large vocabulary 'biased' language models, and therefore can only be applied to languages where these resources exist.

In our own previous work [8, 9, 10], we introduced a lightly supervised method for automatically aligning speech data with imperfect transcripts that does not rely on such resources. Our method comprises two main components: a GMM-based sentence-level segmentation algorithm, and an alignment step which uses incrementally-trained grapheme-level acoustic models to determine the correct orthographic transcription of the segmented utterances. Both steps are lightly supervised, in the sense that they need only small amounts of manual initialisation before proceeding in a fully automatic way with no further intervention from the user, and all statistical models used are learned solely from the speech and text being aligned. Baseline results and evaluations were obtained using a Librivox audiobook recording[1] of *A Tramp Abroad* by Mark Twain, but we have since successfully applied the algorithms to audiobooks in 14 different languages, thus creating the TUNDRA corpus [11].

The success of a speech/text alignment algorithm can be quantified in terms of amount of speech data 'harvested' with correctly aligned transcriptions. While building the TUNDRA corpus, we found that most of the Librivox audiobooks we used had recording conditions that were highly variable within a single book across the different chapters, and that this led to lower harvesting rates. That is, a lower percentage of the data was aligned than expected, especially for chapters where the recording conditions were very different from the book average. Although it can be argued that this noisy data would be better left out of the final speech resource, in many applications the amount of training data is more important than its recording quality and maximising the amount of data aligned is the primary concern. We have therefore been investigating ways to improve the amount of data harvested.

In this paper we apply two adaptation methods to the two main stages of our method: Maximum A Posteriori (MAP) adaptation for the GMM-based segmentation algorithm, and Constrained Maximum Likelihood Linear Regression (CMLLR) transform-based adaptation for the acoustic models (HMMs) used in the alignment step. We show that by employing these techniques, our alignment results for noisy data significantly improve in both the percentage of data aligned

[1]http://librivox.org/a-tramp-abroad-by-mark-twain/ read by John Greenman

and in the accuracy of the aligned data. Although these are standard adaptation procedures, there were some challenges in using them in this context: for MAP, we need to devise a process for selecting the adaptation data in accordance with the specific structure of audiobooks; for CMLLR, the lack of accurate transcripts for the adaptation data, and the use of grapheme-level acoustic models, posed particular problems.

The paper is structured as follows: Sections 2 and 3 present the adaptation methods used for the segmentation and alignment stages respectively. The results obtained with these methods on sets of noisy data are evaluated in Section 4, while Section 5 concludes the paper and discusses future work.

2. Lightly Supervised Speech Segmentation using MAP Adaptation

In [8] we proposed a lightly supervised sentence-level segmentation tool based on Gaussian Mixture Models (GMM) which is an extension of a method widely used for Voice Activity Detection (VAD). The core idea was to train two GMMs: one from the speech segments, and the other from the silence segments, of an initial manually-labelled data set of only 10 minutes of speech. The GMMs were then used to estimate the log likelihood of all segments of the full data being silence or speech. Because short silent pauses can occur within running speech, the algorithm was tuned to detect only sentence boundaries, and not within-sentence pauses. A threshold for discriminating between short pauses and silence was automatically calculated by fitting two Gaussians (one for extended silence and one for short pauses) to the durations of these two types of silent sections, using the manually-labelled data. Results showed a 96% accurate detection rate. Another aspect of performance that we evaluated was the effect of this VAD-based segmentation on the final quality of synthetic voices built from this data. By training two text-to-speech systems, one with a GOLD standard (i.e., manually verified and corrected) segmentation and one with the VAD-based one, we determined that the VAD-based voice had only a marginally, and statistically insignificantly, lower quality.

While the above techniques work well on consistent, clean speech, when used on the data being prepared for the TUNDRA corpus it was found that using GMMs trained only on the small set of manually-labelled data did not give good performance across all the remaining data. This was because they did not capture the correct distributions for silence and speech in the varying noise environments and speaking styles. Therefore, we propose a method to adapt the initial GMMs on a chapter-by-chapter basis. The workflow employed in performing this adaptation and segmentation is presented in Figure 1 and comprises the following steps:

1. **Initial training** – initial GMMs for speech and silence are trained on the labelled data;
2. **1st decoding** – label the speech and silence parts of all chapters using these initial GMMs;
3. **Data selection** – apply a confidence measure to each such speech or silence part, selecting only the *confident* data;
4. **MAP adaptation** – adapt the GMMs using a standard MAP algorithm [12, 13, 14, 15] to this data;
5. **2nd decoding** - re-label the speech and silence parts of all chapters using the adapted GMMs. Segment the chapter at every silence mid-point.

The data selection step described above is used to select the speech and silence segments which are considered to be confidently-labelled and thus suitable as adaptation data. The

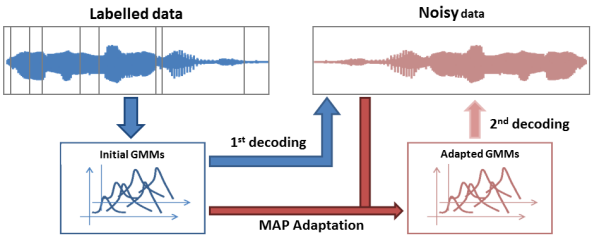


Figure 1: Overview of the MAP adaptation method for the GMM-based VAD.

confidence score or measure we use is based on a log likelihood ratio (LLR) computed for each segment against the respective GMM. Figure 2 shows an example of these histograms for the speech and silence parts of the data corrupted with *babble talk noise at 35dB*: (a) represents the LLR histogram without adaptation; (b) is the histogram after performing MAP adaptation on all the speech and silence parts resulting from the 1st decoding; and (c) is the histogram after MAP adaptation using only the confident segments. This shows that performing adaptation using all the data from the 1st decoding leads to mis-classification of audio segments: the discriminative power of the GMMs is reduced by 18% (compare distance between peaks in Figure 2 (a) with (b)). On the other hand, if data selection is carried out, the histogram plots show an increase in the distance between the average LLRs by 11% (Figure 2 (b) with (c)).

3. Speech Transcription Alignment using CMLLR Adaptation

[9] introduced a lightly supervised and low-resource method for sentence-level alignment of speech with imperfect transcripts. The method incrementally trains grapheme-level acoustic models on the available speech and text data, starting from an initial 10 minute manual orthographic speech transcription[2]. In order to alleviate the consequences of having rather poor acoustic models, the Viterbi decoder was highly restricted by using a so-called *skip network*. The network allows the speech to be matched to any point within an estimated broad text window, but constrains the output to be a consecutive sequence of words from it. To deal with audio deletions, a more relaxed skip network, called a *3-skip* network, can be used which allows a maximum 2 word skip within the hypothesised sequence. To prevent unwanted skips, a bigram language model built from the available text was also used to limit to some extent the 3-skip network. The confidently aligned utterances were then obtained by comparing the recognition acoustic scores using the different types of skip networks. These utterances were then used to retrain the acoustic models, and the process repeated for a couple of iterations. Results from this method showed a 54.1% aligned percentage with 7.64% SER (sentence error rate) and 0.5% WER (word error rate).

Following this, in [10] we increased the alignment percentage by almost 40% (relative) through the use of context-dependent tri-grapheme models and MMI discriminative training. The confidently aligned data then amounted to 75%, with similar sentence and word error rates as in the previous work.

Despite this good performance on our test audiobook (for which we have the GOLD standard alignments required to com-

[2]The same 10 minutes of labelled speech data for the VAD can be used for the aligner as well

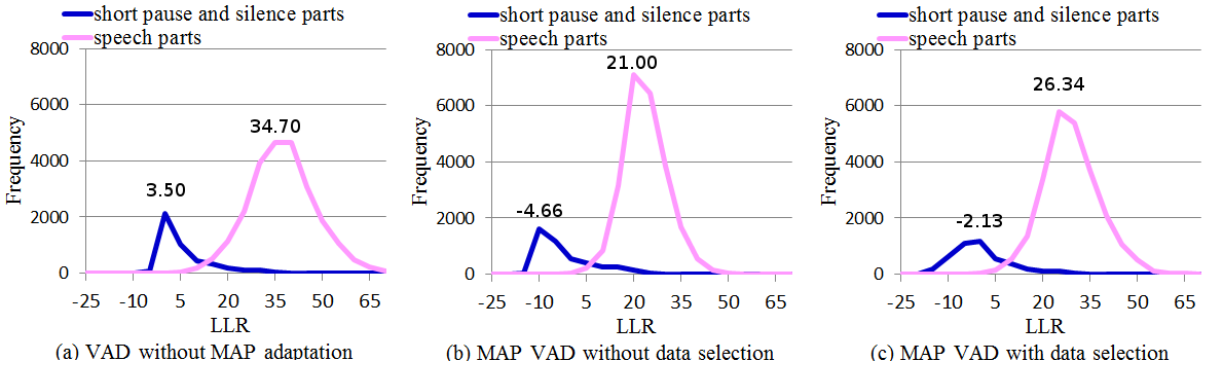


Figure 2: Segment LLR histograms for silence and speech data calculated for (a) VAD without MAP adaptation, (b) MAP VAD without data selection and (c) MAP VAD with data selection. The data on which they are estimated was corrupted with *babble talk noise at 35dB*.

pute SER and WER), when applying the above procedures to other audiobooks from Librivox, we found that the variable recording conditions across chapters (e.g. more background noise as a result of variable distance from the microphone, or worse room acoustics) caused the aligned percentage to drop below 40% for the worst chapters.

To address this problem, we turn to adaptive training methods commonly used in automatic speech recognition and in this paper we propose the use of CMLLR [16], originally proposed for speaker adaptation, for adaptation to the varying channel or environmental conditions found in audiobooks. The CMLLR technique estimates a set of linear transforms for each condition—shared between multiple Gaussians—in a maximum likelihood fashion, making it robust to estimation when the initial transcripts are poor, and allowing effective use of limited adaptation data.

Here we apply CMLLR adaptation to the discriminatively-trained tri-grapheme acoustic models presented in [10] to adapt to the noisy data. Although the poor recognition accuracy over the noisy speech means that the quality of the adaptation transcripts is also quite low, the results show that by using only one or two chapters as adaptation data, the SER and the percentage of aligned data are substantially improved.

4. Results

4.1. Simulated Noisy Speech Recordings

To test our approach, it is necessary to be able to compute SER and WER, for which we need GOLD transcripts. We therefore once again used the Librivox audiobook *A Tramp Abroad* by Mark Twain and degraded two chapters of the audiobook (approx. 28 minutes of speech) by adding noise and/or reverberation to simulate the noisy data found in real recordings in a controllable way.

Through informal evaluation, we determined which conditions approximated those observed in the TUNDRA corpus and similar found data. For reverberation, we convolved the speech with the impulse response of a domestic living room, taken from the Open Air Library [17]. The background noise conditions were replicated using either 8-talker babble or white noise at the following signal-to-noise ratios (SNRs): 10, 15, 20, 25, 30, 35 and 40dB. A total of 29 testing scenarios were obtained this way: reverb; babble noise at each SNR; white noise at each SNR; reverberation and babble noise at each SNR; reverberation and white noise at each SNR. Although 10dB and 15dB SNRs are highly unlikely (i.e., very noisy) for audiobook recordings, these scenarios were kept as points of comparison to evaluate the adaptive power of the acoustic models even when the accuracy of the transcripts is very low.

4.2. GMM VAD with MAP adaptation

We present the evaluation of three versions of VAD: without MAP adaptation; with MAP adaptation, but without data selection; and with MAP adaptation and with data selection. The CORR measure is computed as [18]:

$$CORR = 100 - (FEC + MSC + OVER + NDS) : \quad (1)$$

where the right hand side measures represent (as percentages):

- **FEC** - Front End Clipping - speech classified as silence when passing from silence to speech;
- **MSC** - Mid Speech Clipping - speech classified as silence within a speech sequence;
- **OVER** - silence interpreted as speech at the end of a speech segment;
- **NDS** - Noise Detected as Speech - silence interpreted as speech within a silence part.

Figure 4 shows the CORR measure for each environment. The results show a high dependency on the type of noise and the SNR. For white noise, MAP adaptation gave great improvements at high SNRs. At low SNRs, because of the fact that the initial GMMs were unable to discriminate between speech and silence—all segments were labelled as speech—there are no differences in the CORR measure for the 3 VAD types. The high value of the CORR measure is a result of the fact that the speech segments are much longer than the silence segments, and this influences the FEC and MSC values.

For babble noise, there are noticeable advantages of using MAP, but only for mid-range SNR values. At low and high SNRs, the CORR value is similar to that when no adaptation is used.

When adding reverberation to the clean data, MAP adaptation performs better without the data selection step. This is also true in the case of reverberation and babble talk noise. This may be due to mismatch of the threshold for data selection in these environments. We used the threshold which was the most appropriate for 35dB of babble noise across all environments.

In contrast, VAD without MAP adaptation showed higher CORR across all SNRs for reverberation plus white noise. But

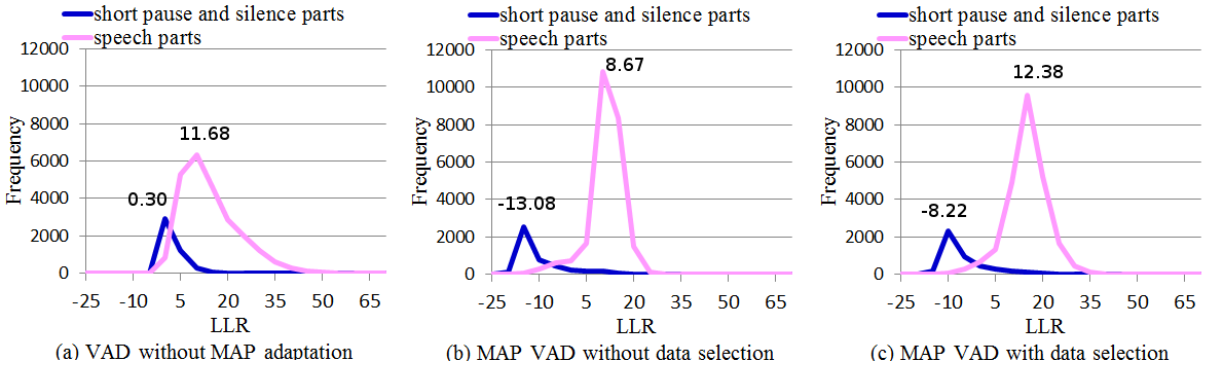


Figure 3: Segment LLR histograms for silence and speech data calculated for (a) VAD without MAP adaptation, (b) MAP VAD without data selection and (c) MAP VAD with data selection. The data on which these examples were estimated was corrupted with *reverberation and white noise at 15dB.*

when examining the LLR histogram for this condition (see Figure 3), adaptation seems increase the discriminative power of the GMMs. This leads us to believe that white noise plus reverberation has the most damaging effect on the GMM-based VAD, and that alternative methods for dealing with this type of scenario must be investigated.

4.3. CMLLR Acoustic Model Adaptation

As described in Section 3, the baseline acoustic models built on the clean data were adapted using the simulated noisy speech data. The adaptation transcripts were obtained from the baseline models using a 1-skip network. For each noisy scenario, we computed the SER and WER of the entire noisy data, as well as the amount of confident data obtained (i.e., the percentage aligned) with its corresponding SER. WERs for the confident data are on average below 1%, do not seem to be influenced by the adaptation step, and are therefore not presented.

Figures 5 and 6 present the SER and WER of the entire noisy data respectively. The SER and WER values are computed for the text aligned using the adapted acoustic models, as compared to the GOLD standard transcripts. As expected, SER and WER are reduced by the use of adaptation, especially at low SNRs. The type of noise has a strong influence on the overall performance: white noise in conjunction with reverberation has the most damaging effect on the performance of the clean acoustic models. One other thing worth noting is the fact that, although the SER in some cases is quite high, the corresponding low WER makes adaptation possible. For example in the case of white noise at 20dB SNR, the SER is around 70%, but the WER is around 20%. The adaptation in this case improves both the WER and SER of the entire noisy data by a substantial amount (approx. 50% for SER and 20% for WER).

The improvement in the SER and WER of the noisy data through adaptation would mean nothing unless it also influences the aligned data percentage. In figure 7 we present this influence. The bars in the figure represent the percent of confident data with its relative SER. Again, the adaptation makes the percent of confident data increase, and also lowers its SER. The average increase in confident data percentage is 20%, with a maximum of 62% for the reverberation and white noise at 25dB scenario. In extreme cases, the adaptation did not help (such as white noise at 10dB and 15dB, reverberation and babble noise 10dB, reverberation and white noise 10dB and 15dB), but these are almost certainly not of interest anyway, if the speech is going to be used to build a speech synthesiser.

Note that the numbers presented in this section are not directly comparable to those in [10], because here we are evaluating only a small subset of the audiobook, and not its entirety.

5. Conclusions

In this paper we have shown the advantages of using adaptive techniques in order to improve the alignment accuracy of text with corresponding noisy and/or reverberated speech, which for experimental purposes was created by simulating the conditions we have observed in various typical non-professional audiobooks.

The speech segmentation algorithm performance is highly dependent on the noise characteristics, giving variable improvements across the tested scenarios. The presence of reverberation leads to unexpected results in terms of the CORR measure, and the white noise plus reverberation renders the adaptation ineffective. On the other hand, when applying adaptation to the acoustic models of the speech aligner, the amount of confident data increases in all scenarios, resulting in an average 20% improvement. It also reduces the SER of this data.

Future work includes the evaluation of the effect of adaptation for both segmentation and alignment on the confident data percentage. Another technique which can be employed is to cluster data based on recording environments, and do cluster-based adaptation (rather than chapter-based). We would also like to investigate the influence of the VAD indices (CORR, FEC, MSC, OVER and NDS) on the TTS system's quality when using various noise environments and amount of adaptation data.

6. Acknowledgements

The GOLD transcripts for *A Tramp Abroad* were very kindly provided by Toshiba Research Europe Limited, Cambridge Research Laboratory. The research leading to these results was partly funded from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement 287678 (Simple4All), EPSRC EP/I031022/1 (NST) and EP/J002526/1 (CAF). The research presented here has made use of the resources provided by the Edinburgh Compute and Data Facility (ECDF: `http://www.ecdf.ed.ac.uk`). The ECDF is partially supported by the eDIKT initiative (`http://www.edikt.org.uk`).

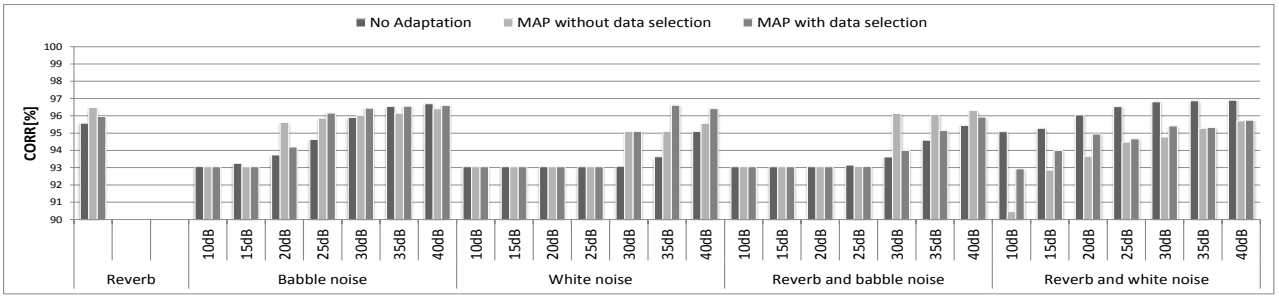


Figure 4: CORR measure for each VAD method: using no adaptation, with MAP adaptation but without data selection and with MAP adaptation and data selection.

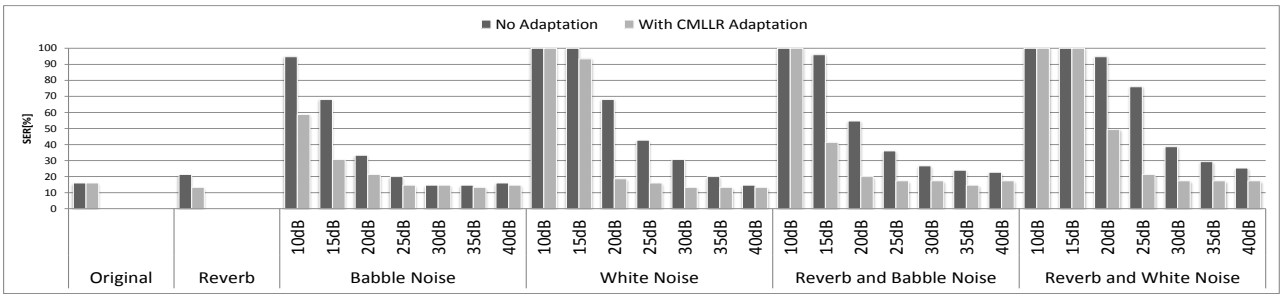


Figure 5: SER for each noisy and reverberant data set, with and without CMLLR adaptation. SER is computed on the retrieved text for each acoustic model against a GOLD standard transcription.

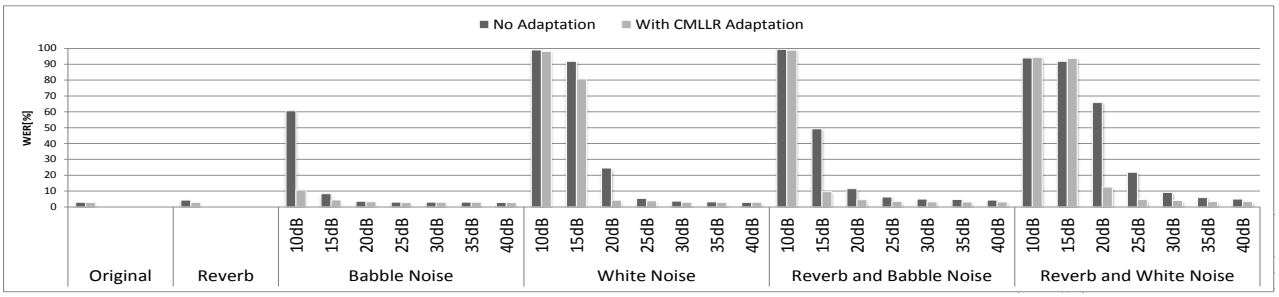


Figure 6: WER for each noisy and reverberant data set, with and without CMLLR adaptation. WER is computed on the retrieved text for each acoustic model against a GOLD standard transcription.

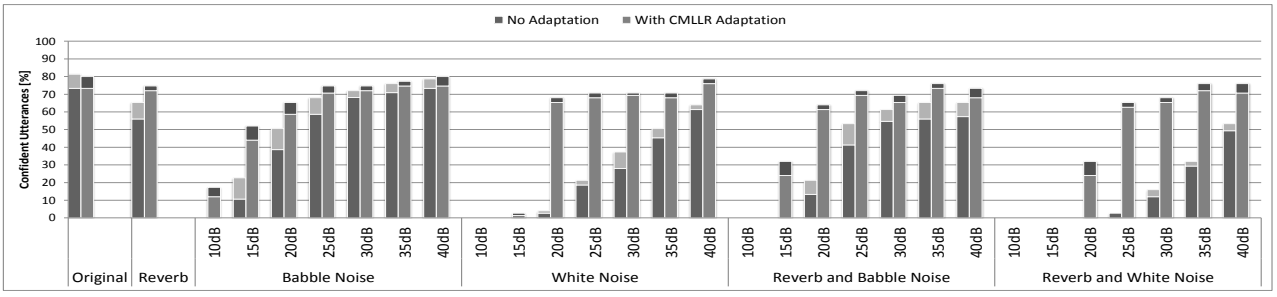


Figure 7: Aligned data percentage obtained before and after CMLLR adaptation. The different colour bar at the top of each column represents the relative SER for each data set.

7. References

[1] É. Székely, T. G. Csapó, B. Tóth, P. Mihajlik, and J. Carson-Berndsen, "Synthesizing expressive speech from amateur audiobook recordings," in *Proc. IEEE Workshop on Spoken Language Technology*, Miami, Florida, USA, 2012, pp. 297–302.

[2] O. Boeffard, L. Charonnat, S. L. Maguer, and D. Lolive, "Towards Fully Automatic Annotation of Audio Books for TTS," in *Proc. of LREC*, Istanbul, Turkey, may 2012.

[3] N. Braunschweiler, M. Gales, and S. Buchholz, "Lightly supervised recognition for automatic alignment of large coherent speech recordings," in *Proc. of Interspeech*, 2010, pp. 2222–2225.

[4] K. Prahallad, A. R. Toth, and A. W. Black, "Automatic building of synthetic voices from large multi-paragraph speech databases," in *Proc. of Interspeech*, 2007, pp. 2901–2904.

[5] P. Moreno and C. Alberti, "A factor automaton approach for the forced alignment of long speech recordings," in *Proc. of ICASSP*, 2009, pp. 4869–4872.

[6] G. Bordel, M. Peñagarikano, L. J. Rodríguez-Fuentes, and A. Varona, "A simple and efficient method to align very long speech signals to acoustically imperfect transcriptions," in *Proc. of Interspeech*, 2012.

[7] M. Alessandrini, G. Biagetti, A. Curzi, and C. Turchetti, "Semi-Automatic Acoustic Model Generation from Large Unsynchronized Audio and Text Chunks," in *Proc. of Interspeech*, 2011, pp. 1681–1684.

[8] Y. Mamiya, J. Yamagishi, O. Watts, R. A. J. Clark, S. King, and A. Stan, "Lightly Supervised GMM VAD to use Audiobook for Speech Synthesiser," in *Proc. ICASSP*, 2013.

[9] A. Stan, P. Bell, and S. King, "A grapheme-based method for automatic alignment of speech and text data," in *Proc. IEEE Workshop on Spoken Language Technology, Miami, Florida, USA*, 2012.

[10] A. Stan, P. Bell, J. Yamagishi, and S. King, "Lightly Supervised Discriminative Training of Grapheme Models for Improved Sentence-level Alignment of Speech and Text Data," in *Proc. of Interspeech (accepted)*, 2013.

[11] A. Stan, O. Watts, Y. Mamiya, M. Giurgiu, R. A. J. Clark, J. Yamagishi, and S. King, "TUNDRA: A Multilingual Corpus of Found Data for TTS Research Created with Light Supervision," in *Proc. of Interspeech (accepted)*, 2013.

[12] Y. Zhang and M. S. Scordilis, "Effective online unsupervised adaptation of gaussian mixture models and its application to speech classification," *Pattern Recognition Letters*, vol. 29, no. 6, pp. 735–744, 2008.

[13] J. Gauvain and C. Lee, "Maximum a posteriori estimation for multivariate gaussian mixture observations of Markov chains," *IEEE Transactions on Speech and Audio Processing*, vol. 2, no. 2, pp. 291–298, 1994.

[14] T. Oonishi, K. Iwano, and S. Furui, "Noise-robust speech recognition decorder using speech/non-speech confidence measures," *Technical Report of IEICE*, vol. 110, no. 81, pp. 49–54, 2010.

[15] D. A. Reynolds, T. F. Qatieri, and R. B. Dunn, "Speaker verification using adapted gaussian mixture models," *Digital Signal Processing*, vol. 10, no. 1, pp. 19–41, 2000.

[16] M. Gales, "Maximum likelihood linear transformations for HMM-based speech recognition," *Computer Speech and Language*, vol. 12, pp. 75–98, 1998.

[17] S. Shelley, A. Foteinou, and D. Murphy, "OpenAIR: An Online Auralization Resource with Applications for Game Audio Development," in *Proceedings of the 41st Int. Conference of the AES, Audio for Games*, 2011.

[18] F. Beritelli, S. Casale, and G. Ruggeri, "Performance evaluation and comparison of ITU-T/ETSI voice activity detectors," in *Proc. ICASSP*, vol. 3, Salt Lake City, UT, USA, 2001, pp. 1425–1428.

Speech synthesis using a maximally decimated pseudo QMF bank for embedded devices

Nobuyuki Nishizawa and Tsuneo Kato

KDDI R&D Laboratories Inc., Japan

{no-nishizawa, tkato}@kddilabs.jp

Abstract

A fast speech waveform generation method using a maximally decimated pseudo quadrature mirror filter (QMF) bank is proposed. The method is based on subband coding with pseudo QMF banks, which is also used in MPEG Audio. In the method, subband code vectors for speech sounds are synthesized from magnitudes of spectral envelope and fundamental frequencies for periodic frames, and then waveforms are generated by decoding of the vectors. Since the synthesizing of the vectors is performed at the reduced sampling rate by the maximal decimation and the decoding is processed with fast discrete cosine transformation algorithms, faster speech waveform generation is achieved totally. Although pre-encoded vectors for noise components were used to reduce the computational costs in our former studies, in this study, all code vectors for noise components are made with a noise generator at run time for small footprint systems. In contrast, a subjective test for synthetic sounds by HMM-based speech synthesis using mel-cepstrum showed the proposed method was comparable to our former method and also the conventional method using a mel log spectrum approximation (MLSA) filter in quality of sounds.

Index Terms: HMM-based speech synthesis, speech waveform generation, filter bank, subband coding, embedded systems

1. Introduction

HMM-based speech synthesis [1, 2] is suitable for embedded devices since it can generate high-quality sounds with small footprints such as several hundred kilobytes or megabytes. However, in the computational cost on the devices, the HMM-based speech synthesizers are often inferior to concatenative speech synthesis with a small waveform segment database optimized for embedded systems. This is because waveform generation in the HMM-based speech synthesis is performed by means of costly signal processing like the mel log spectrum approximation (MLSA) filters [3] in which several hundreds multiply-accumulate operations per output sample are required.

As a more efficient method for waveform generation, we studied speech synthesis using filter banks performed at the reduced sampling rate [4, 5]. In the studies, the pseudo quadrature mirror filter (QMF) banks [6, 7], which are also used in MPEG Audio [8], have been the main focus because faster processing is possible with fast Fourier tansformation (FFT) or similar fast discrete cosine transformation such as Lee's DCT [9]. Our previous method [5] was based on a filter bank-based speech synthesis. In the method, white source waveforms such as noise sequences or impulses were initially decomposed into several bands by a set of band-pass filters. Then, the amplitudes of the decomposed (i.e. band-limited) waveforms were scaled for each band to build spectral features. Finally the decomposed and scaled waveforms were composed into speech waveforms by simple summation. These operations were performed on the subband domain in the subband coding system. Thus, in the method, decomposed source waveforms were initially subband-coded. The scaling and the summation were performed against the code vectors at the reduced sampling rate, not waveforms. Since the code vectors from band limited waveform could be sparse, the computational cost for the waveform generation can be reduced even though the cost for decoding of the code vectors is taken into account. Moreover, sinusoidal synthesis [10] performed in the subband domain was also introduced for accurate reproduction of spectra, where the amplitude of each harmonic component is directly configurable independent of the filter bank-based speech synthesis. By these techniques, fast processing without degradation in quality of sounds was achieved.

However, for fast processing, use of predecomposed and pre-encoded source waveforms was expected in the method. Although it removed the filter banks for the source waveform decomposer and encoder from speech synthesizers, it instead required large storage for the predecomposed and pre-encoded code vectors of impulses and noises. Therefore, in this study, an improved method to generate code vectors without code vector storage is proposed. In the proposed method, aperiodic components are directly constructed by the pseudo QMF bank from

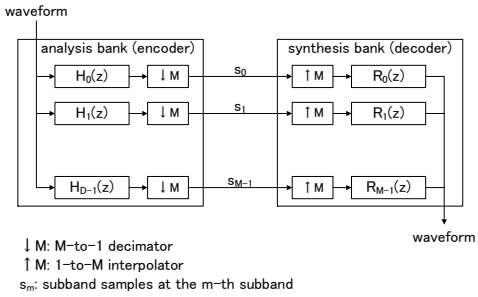


Figure 1: Block diagram of the subband coding system based on the maximally decimated filter banks.

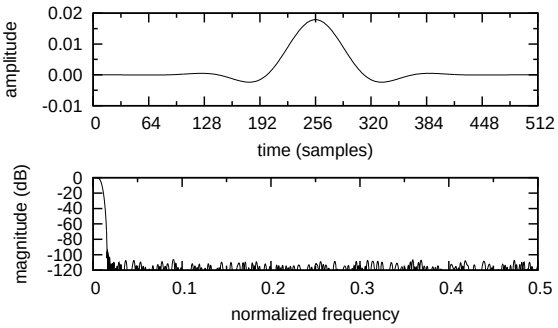


Figure 2: Impulse response and magnitude-frequency response of the prototype filter for a 32-band pseudo QMF bank.

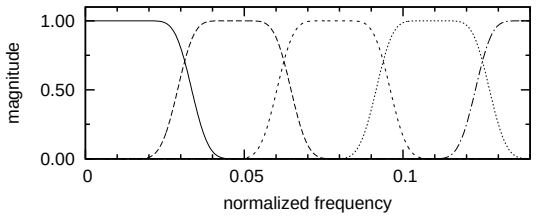


Figure 3: Magnitude-frequency response of the filters for 0th (lowest) to 4th bands in the analysis bank. (The filters correspond to $H_0(z)$ to $H_4(z)$ in Figure 1.)

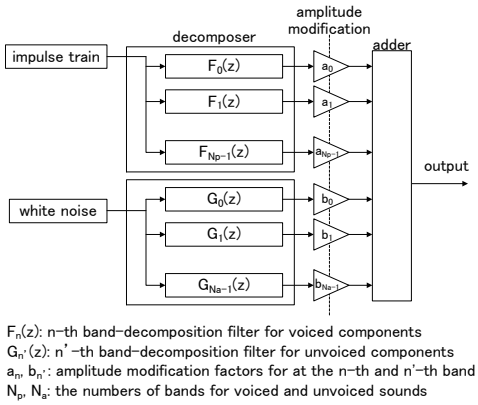


Figure 4: Block diagram of the filter bank based speech synthesizer.

white noise at run time. In contrast, all periodic components are synthesized by sinusoidal synthesis, which is similar to our former method. The result of a subjective test indicates that the proposed method can generate speech waveforms with quality comparable to those of our former method and the conventional MLSA filter-based method.

The rest of the paper is organized as follows. First, in Section 2, our former waveform generation method based on filter banks is introduced. Section 3 explains the proposed method. Section 4 gives a subjective test to evaluate degradation in speech sound quality by the proposed method. Finally, section 5 concludes the paper.

2. Waveform generation performed on subband coding

2.1. Subband coding based on maximally decimated pseudo QMF bank

Figure 1 shows a block diagram of a subband coding system with maximally decimated filter banks where the number of subbands is M. In the system, input waveforms are equally decomposed into subbands in the analysis bank, and then the decomposed waveforms are composed in the synthesis bank.

Commonly, analysis and synthesis filters of pseudo QMF banks are made by cosine modulation of the impulse response of a prototype filter. The cosine modulation corresponds to shifting of the magnitude response along the frequency axis. In this study, the impulse responses of the filters of the analysis and synthesis banks are given by:

$$h_m(i) = 2h(i)\cos\Big(\frac{\pi}{64}(2m+1)(i-16)\Big) \quad (1)$$

$$r_m(i) = 2h(i)\cos\Big(\frac{\pi}{64}(2m+1)(i+16)\Big) \quad (2)$$

where $h(i)$ is the impulse response of the prototype filter. These equations are from the MPEG Audio specification [8]. In this study, the prototype filter defined in the MPEG Audio specification, which is for 32-band filter banks, is also used. The length of the filters is 512. This configuration is the same as that in our former study [5]. Thus, filter banks that are components of highly optimized MPEG Audio decoders like [11] are directly applicable to systems for the proposed method. Figure 2 shows plots of the coefficients and magnitude-frequency response of the prototype filter, respectively. Figure 3 shows the magnitude-frequency responses of the cosine modulated filters for the analysis bank.

2.2. Filter bank-based speech synthesis on the subband coding system

Figure 4 shows a block diagram of the filter bank-based speech synthesizer. In this system, impulse trains and white noise sequences as source waveforms are initially band-decomposed by filter banks. Then, spectral features of synthetic speech sounds are constructed from the band-decomposed waveforms with amplitude modification. This amplitude modification should be controlled with appropriate delays to compensate delays in the filter bank.

For simplicity of the processing, it is desirable that simple summation of the decomposed waveforms without amplitude modification restores the original source waveform. Therefore, cosine modulated filters of the Nth-band filter [12] are used for the band-decomposition. The N-th band filter is a linear-phase low-pass filter where the edge of the stopband is $1/2N$ in normalized frequency, and the magnitude responses at 0 and $1/4N$ in normalized frequency are approximately 1 and $1/2$, respectively. In this study, N equals 32.

In contrast, the cosine modulation of the filter coefficients corresponds to shifting of the magnitude response along the frequency axis. The cosine modulation for the n-th filter $f_n(i)$ for band-decomposition is performed by the following equation:

$$f_n(i) = 2f(i)\cos\Big(\frac{\pi}{2N}(2n+1)i\Big)\ (0 \le n \le N-1) \quad (3)$$

where $f(i)$ denotes the impulse response of the prototype N-th band filter. In this configuration, the edges of the n-th passband are $(2n-1)/4N$ and $(2n+1)/4N$ in normalized frequency.

Thus, a filter bank for white source waveform decomposition can be built where the summation of all outputs of bands approximately performed the all pass characteristic:

$$|F_n(\omega) + F_{n+1}(\omega)| \approx 1 \quad (4)$$

$$\text{for } \frac{\pi}{N}(n+\frac{1}{2}) \le \omega \le \frac{\pi}{N}(n+\frac{3}{2}),\ 0 \le n < N-1$$

Consequently, the magnitude at ω is given as:

$$\begin{aligned}&|a_n F_n(\omega)S(\omega) + a_{n+1}F_{n+1}(\omega)S(\omega)| \\ &\quad = a_n|F_n(\omega)| + a_{n+1}|F_{n+1}(\omega)| \end{aligned} \quad (5)$$

where a_n and $S(\omega)$ denote the amplitude modification factor for n-th band and the magnitude response of a white noise, respectively, and $|S(\omega)| = 1$.

Of course, the filter bank-based method increases the computational cost due to processing for multiple bands. As a method to reduce the computational cost, sampling rate reduction with the bandwidth limitation was studied. Consequently, the filter bank-based speech synthesizer was implemented on the subband coding system based on the pseudo QMF bank in our former study [5]. In the method, encoding and decoding of the subband coding were performed after the band decomposition and after the composition of the bands. Since predecomposition and pre-encoding of the white source waveforms are adopted, there is neither filter bank for the source decomposition nor encoder for the subband coding.

Subband-coded vectors for band limited waveforms can be sparse. Elements of the coded vectors corresponding to the overlapped region of the stop bands of bandpass filters for source waveform decomposition and encoding of the subband coding can be regarded as zero since all components are cut in the region. For such sparse vectors, the computational cost for the scaling to construct spectral features is not high. Although multiple elements in the coded vector were non-zero values due to the overlapped structure of the filter bank in the encoder, affection of sampling rate reduction by the subband coding is more significant. Consequently, the computational cost can be reduced by adoption of the subband coding.

2.3. Calculation algorithm for amplitude modification factors from the mel-cepstrum

Since the feature vector of the target system is the mel-cepstrum, the amplitude modification factors for subbands are extracted from the mel-cepstra. To reduce error in this conversion, more dimensions in the intermediate vectors of the conversion than the order of mel-cepstrum should be used. Consequently, this dimension can become the main contributor to the computational complexity of speech synthesis.

For this conversion, initially, the mel-cepstrum for each frame is converted into a log-power in the mel-scale spectrum by a DFT or DCT operation. Then, at the center of each subband in the mel frequency scale, the log-power coefficient is extracted from a log-spectral envelope built by linear interpolation. Next, the log power coefficients, which equal the amplitude modification factors where the power of source is normalized, are converted into linear power coefficients. Power operations in this conversion can be effectively calculated with table lookup, shift operation and linear interpolation.

In the above algorithms, the worst operation in terms of computational complexity is DFT (or DCT), the complexity of which can be $O(N \log N)$ per frame, i.e. $O(\log N)$ per sample.

2.4. Sinusoidal synthesis in the subband domain

In our first study [4], spectral features of voiced sounds were constructed by the scaling of each element of coded vectors. However, insufficiency of the resolution along the frequency axis caused degradation of the quality of the synthetic sounds. Therefore, speech synthesis by si-

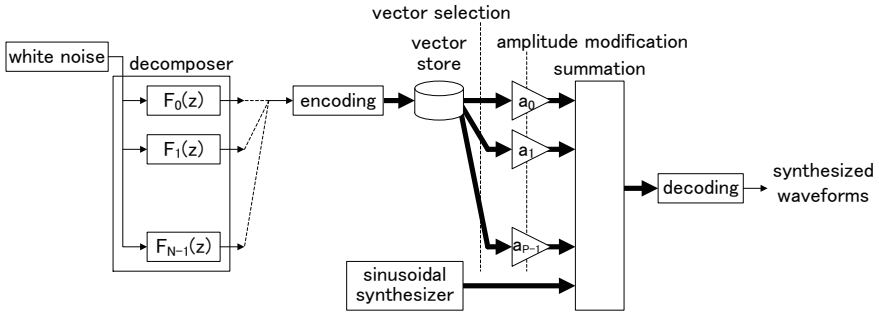


Figure 5: Block diagram of the speech synthesizer proposed in our former study [5]. Bold lines correspond to coded vectors.

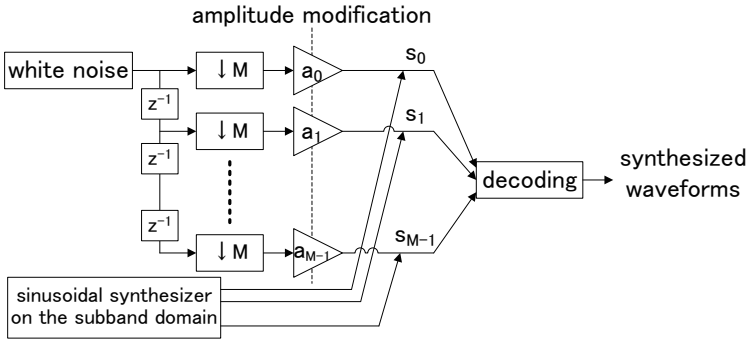


Figure 6: Block diagram of the speech synthesizer based on the proposed method.

nusoids [10] was introduced in our former study [5] to improve the accuracy of the spectral feature reproduction in the proposed method. Since intervals of harmonic components are usually narrower than those of the bands, errors in the spectral feature reproduction are reduced.

In the following, for the sake of simplicity, a formulation is introduced where waveforms are built by the summation of cosine functions rather than sine functions:

$$x(t) = \sum_k A_p(\omega_0) \cos\bigl(\omega_0 k(t - t_0)\bigr) \qquad (6)$$

where ω_0, $A_p(\omega_0)$ and t_0 are angular frequency of fundamental vibration, amplitude of the target sound at ω_0 and the position of the corresponding impulse, respectively.

Since magnitude-frequency and phase-frequency responses of the filters of the analysis bank are given, encoded results of sinusoids are easily obtained by calculations in the frequency domain. Where $|H_m(\omega)|$ and $\arg H_m(\omega)$ are magnitude-frequency and phase-frequency responses of the analysis filter for the m-th subband, respectively, the m-th element of the subband vector of encoded cosine waveform with angular frequency ω is given as follows:

$$x_{\omega,m}(t) = |H_m(\omega)| A_p(\omega) \cos\bigl(\omega(t - t_0) + \arg H_m(\omega)\bigr) \qquad (7)$$

where $A_p(\omega)$ denotes the magnitude of the periodic component at angular frequency ω.

Referring to Eq. (1), $\arg H_m(\omega) = -\pi(2m + 1)/4$. In contrast, $|H_m(\omega)|$ is easily obtained with a precalculated table. For example, a 512-entry table for between 0 and $\pi/32$ in angular frequency was used with shift and reverse operations along the frequency axis and linear interpolation to synthesize sounds for the following evaluation.

Basically, one sinusoid encodes into two subbands due to the overlapping structure of the analysis bank (refer to Fig. 3). By these operations, the sinusoidal synthesis is performed in the subband domain with the reduced sampling rate in the maximally decimated filter bank.

Consequently, the system shown in Fig. 5 was introduced in our former study.

3. Run-time generation of code vectors for aperiodic components

In the proposed method, elements of vectors for aperiodic components are generated on demand. In other words, aperiodic components are built by the combination of subband code vectors each of which has only one nonzero element. Referring to Fig. 1, in the first step of decoding of subband code vectors, bandwidth expansion is performed, then the expanded components are filtered by the band-pass filters in the decoder. However, different from the decoding of common subband coding based on the maximally decimated QMF banks, no alias cancel-

ing is taken into account for aperiodic components in the proposed method.

Now, two neighboring elements in the code vector are focused on in the following discussion. Commonly, prototype filters for pseudo QMF banks are designed with the condition where squared magnitude responses of analysis and synthesis banks approximately equal the magnitude responses of the N-th band filter to achieve approximately perfect reconstruction of the input. The prototype filter for the MPEG Audio also has similar properties. Therefore, in the subband coding system based on pseudo QMF banks, summation of the squared responses of neighboring subbands that overlap each other has an approximately all-pass characteristic:

$$\begin{aligned} &|H_m(\omega)R_m(\omega) + H_{m+1}(\omega)R_{m+1}(\omega)| \\ &\quad = |R_m(\omega)|^2 + |R_{m+1}(\omega)|^2 \approx 1 \end{aligned} \tag{8}$$

$$\text{for } \frac{\pi}{M}(m+\frac{1}{2}) \leq \omega \leq \frac{\pi}{M}(m+\frac{3}{2}),\ 0 \leq m < M-1$$

where $H_m(\omega)$ and $R_m(\omega)$ are the magnitude responses of the filters for the m-th subband in the analysis and synthesis bank, respectively. In general, when two elements are set from independent white noises, the variance of summation of the two elements equals the summation of the variances of the two elements. Thus, the magnitude at ω is given with independent white noises as:

$$\begin{aligned} &|a_m R_m(\omega)S_m(\omega) + a_{m+1}R_{m+1}(\omega)S_{m+1}(\omega)| \\ &\quad = (a_m^2|R_m(\omega)|^2 + a_{m+1}^2|R_{m+1}(\omega)|^2)^{1/2} \end{aligned} \tag{9}$$

where $S_m(\omega)$ denotes an independent white noise and $|S_m(\omega)| = 1$. Where $a_m = a_{m+1}$, spectral features become flat similar to our former method.

Thus, vectors for aperiodic components can be built with an independent white noise. Figure 6 shows a block diagram of the speech synthesizer based on the proposed method. Structure with delays and down-samplers in Fig. 6 corresponds to sequentially value assignment into elements of the coded vectors.

For example, in our former system [5], a cyclic quasi noise series for which the period was 4096 samples (128 frames) was used for aperiodic components. Since only neighboring 3 elements rather than M $(= 32)$ elements for each vector of predecomposed bands were required to be stored for aperiodic components, the required size of the table is 12288 $(= 32 \times 3 \times 128)$ for the 32-band predecomposition. In contrast, the table is unnecessary for the proposed method. Consequently, in the proposed method, tables for cosine function for the sinusoidal synthesis, the filter coefficients and magnitude response of the 512-tap prototype filter for the pseudo QMF bank are necessary.

However, although the filter banks for the source decomposition and the subband coding became separated in our former system for flexibility of the source decomposition, the decomposition of noise in the proposed method depends on the design of the pseudo QMF bank again. Nevertheless, the result of the experiment in our former study where the sinusoidal synthesis was adopted implied that error in the reproduction of the spectral feature especially for low band was subjectively problematic only for periodic components since the frequency resolution in the aperiodic source decomposition was comparable to that in the subband coding.

4. Subjective evaluation

In practice, quality of speech sounds should be at least comparable to those of the conventional methods even when faster and smaller waveform generation is achieved. Although the difference between our proposed and former methods is limited only in aperiodic components, to evaluate the quality of sounds subjectively,a mean opinion score (MOS) test of the synthetic sounds was conducted.

In the test, the synthesis targets were generated by our HMM-based speech synthesizer using melcepstrum for a male and female voice. HMMs were trained from 6.1-hour and 5.8-hour sounds for the male and female voice, respectively. For spectral features, 39-order melcepstrum was used where the warping factor $\alpha = 0.4375$ $(= 7/16)$. All voiced sounds were synthesized only from periodic components. This corresponds to excitation only by impulse trains in the conventional methods. In the test, 10 subjects listened to synthetic speech sounds and scored them on a 5-point discrete scale (1: very poor, 2: poor, 3: fair, 4: good, 5: very good) to express their preferences. In this test, the sampling rate was 16 kHz. Although the frame period of the HMMs was 5 ms, that in the waveform generation by the proposed method was 2 ms (32 samples) because the number of subbands was fixed at 32. This conversion was performed with linear interpolation. Subband amplitude modification factors were determined at the center frequencies of the bands, and extracted from the extracted melcepstrum through power spectra in the mel-scale with linear interpolation, where the number of dimensions for mel-spectrum was 64 for all conditions. For comparison, we also prepared speech sounds synthesized using a 39-order MLSA filter, and our former filter bank-based method using pre-encoded vectors where the number of subbands was 32. Although impulse trains were also used to generate periodic components in our former study [5], all periodic components for this test were synthesized from sinusoids.

For each condition, stimuli for 10 sentences that were similar to those from J01 to J10 of the ATR503 corpus [13] were prepared. The stimuli were randomly ordered for each subject and presented to both ears through closed-ear headphones in a silent room.

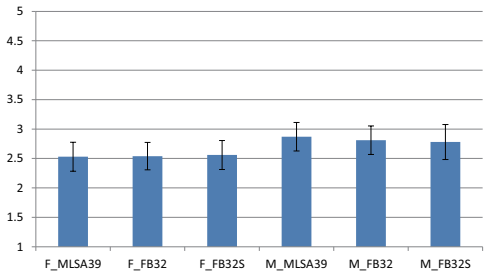


Figure 7: Result of the subjective evaluation. In the conditions, prefix F and M correspond to female and male sounds, and suffix MLSA39, FB32, and FB32S correspond to the conventional method using a 39-order MLSA filter, the conventional 32-band filter bank-based method with pre-encoded vectors and the proposed method, respectively. The error bars indicate the 95% confidence intervals.

Figure 7 shows the results of the test. The scores were comparable among all conditions for each of the male and female voices, i.e., the proposed method can be a substitute for the waveform generation method. Although approximatey 0.3 point differences were observed between the male and female voices, they would depend only on the HMMs because of the comparable results for all conditions in each voice.

It should be noted that the method based on filter banks for resynthesized speech with parameters extracted from natural speech sounds was subjectively superior to the method using the MLSA filter in the former study [5]. The result could be caused by fluctuation of the filter parameters estimated from natural speech sounds; MLSA filters with insufficient margins in Padé approximation can be unstable temporarily, especially when the parameters vary quickly. By contrast, the proposed method, which is a finite impulse response system, is stable anytime. Nevertheless, in the HMM-based speech synthesis, smoothed trajectories of the parameters with consideration of delta and delta-delta features are commonly generated [1]. This would be the reason why no difference was observed in this test.

5. Conclusion

This paper presented a waveform generation method using a pseudo QMF bank for embedded devices. In the method, all coded vectors were synthesized at run time with cosine functions for periodic components and a white noise generator for aperiodic components. The results of a subjective test using synthetic speech sounds indicated that the method was comparable to both the conventional methods with an MLSA filter and our former method by a filter bank with pre-encoded vectors in terms of the quality of sounds. Therefore, using the proposed method, speech synthesizers with smaller footprints than those of the conventional systems can be built without degradation in sound quality.

6. References

[1] Masuko, T., Tokuda, K., Kobayashi, T. and Imai, S., “Speech synthesis using HMMs with dynamic features,” in Proc. of ICASSP ’96, vol. 1, pp. 389–392, May 1996.

[2] Yoshimura, T., Tokuda, K., Masuko, T., Kobayashi, T. and Kitamura, T., “Simultaneous modeling of spectrum, pitch and duration in HMM-based speech synthesis,” in Proc. Eurospeech ’99, pp. 2347–2350, Sep. 1999.

[3] Imai, S., “Cepstral analysis synthesis on the mel frequency scale,” in Proc. ICASSP ’83, vol. 8, pp. 93–96, Apr. 1983.

[4] Nishizawa, N. and Kato, T., “Speech synthesis using a non-maximally decimated filter bank for embedded systems,” in Proc. INTERSPEECH 2012, Portland, OR, U.S.A., Wed.O6d.04, Sep. 2012.

[5] Nishizawa, N. and Kato, T., “Speech synthesis using subband-coded multiband source components and sinusoids,” in Proc. ICASSP 2013, Vancouver, Canada, pp. 8002–8006, May. 2013.

[6] Rothweiler, J. H., “Polyphase quadrature filters – A new subband coding technique,” in Proc. ICASSP ’83, Boston, MA, U.S.A., vol. 3, pp. 1280–1283, Apr., 1983.

[7] Princen, J. P., Johnson, A. W. and Bradley, A. B., “Subband/transform coding using filter bank designs based on time domain aliasing cancellation,” in Proc. ICASSP ’87, Dallas, TX, U.S.A., vol. 4, pp. 2161–2164, Apr. 1987.

[8] ISO/IEC, JTC1/SC29/WG11 MPEG, “Information technology – Coding of moving pictures and associated audio for digital storage media at up to about 1.5 Mbit/s Part 3: Audio,” IS11172-3, 1992.

[9] Lee, B. G., “A new algorithm to compute the discrete cosine transform,” IEEE Trans. on ASSP, vol. 32(6), pp. 1243–1245, Dec. 1984.

[10] Quatieri, T. F. and McAulay, R. J., “Speech transformations based on a sinusoidal representation,” IEEE Trans. on ASSP, vol. 34(6), pp. 1449–1464, Dec. 1986.

[11] Hans, M. C. and Bhaskaran, V., “A fast integer implementation of MPEG-I audio decoder,” HP Labs Technical Reports, HPL-96-03, Jan. 1996.

[12] Mintzer, F, “On half-band, third-band, and Nth-band FIR filters and their design,” IEEE Trans. on ASSP. vol. 30(5), pp. 734–738, Oct. 1982.

[13] Abe, M., Sagisaka, Y., Umeda, T. and Kuwabara, H., “Speech Database User’s Manual,” ATR Interpreting Telephony Research Laboratories Technical Report, TR-I-0166, Japan, Aug. 1990 (in Japanese).

HMM-based sCost quality control for unit selection speech synthesis

Sathish Pammi[1], *Marcela Charfuelan*[2]

[1]ISIR, Universit Pierre et Marie CURIE (UPMC), Paris, France
[2]DFKI, Language Technology Lab, Berlin, Germany
sathish.pammi@isir.upmc.fr, marcela.charfuelan@dfki.de

Abstract

This paper describes the implementation of a unit selection text-to-speech system that incorporates a statistical model Cost (sCost), in addition to target and join costs, for controlling the selection of unit candidates. sCost, a quality control measure, is calculated off-line for each unit by comparing HMM based synthesis and recorded speech with their corresponding unit segment labels. Dynamic time warping (DTW) is used to perform such comparison at level of spectrum, pitch and voice strengths. The method has been tested on unit selection voices created using audio book data. Preliminary results indicate that the use of sCost based only on spectrum introduce more variety on style pronunciation but affects quality; whereas using sCost based on spectrum, pitch and voicing strengths improves significantly the quality, maintaining a more stable narrative style.
Index Terms: Text-to-speech synthesis, unit selection synthesis, statistical parametric synthesis, quality control

1. Introduction

Quality control of units in unit selection speech synthesis is a topic of high interest. Especially important are automatic approaches for finding the units that are intelligible and labelling error free for stable and good quality synthesis. Transcription and automatic labelling errors are the most frequent problems in unit selection synthesis. When we are dealing with large audio book corpora, the additional problem is handling the variable expressivity. The narrator in the audio book, might produce such a variability in speech style and pronunciation, that avoiding artifacts and abrupt changes in waveform concatenation is still a matter of research [1].

The use of HMM-based synthesis techniques to improve speech quality in unit selection is not a new topic. Several researchers have attempted to combine in a hybrid approach, statistical prediction of parameters with waveform concatenation. For example in [2, 3] a HMM-based unit selection approach is proposed, where acoustic parameters (spectral and fundamental frequency) generated with HMM models are used to guide the selection of units. This is done via sentence likelihood and a feature vector distance between HMM generated features and extracted features from the waveform unit candidates. A similar approach, using diphones as unit level, is adopted in [4], where a hybrid technique of unit selection from statistically predicted parameters is proposed. Also in [5] normalised distances between HMM trajectory and those of the waveform unit candidates are used for selecting final candidates in a unit sausage (lattice). The main difference in this last case, is an additional pruning strategy to generate a compact set of unit candidates.

In this paper a HMM-based synthesis approach is also used to improve unit selection speech quality. Like in the hybrid approach we use HMM-based trained models to generate acoustic parameters, but here we use those parameters off-line to precalculate a statistical model cost (sCost). Thus, the sCost is a measure of how different a sentence of the corpus is (in terms of acoustic parameters at level of units) from a sentence generated with statistically trained models (HMMs).

The sCost measure was developed in our previous work [6], where it was used to automatically find labelling errors, so to improve the quality of concatenation units. In this paper we extend our previous work in two ways: (i) sCost is used in addition to target and join costs for controlling the selection of unit candidates in a unit selection synthesiser; and (ii) sCost is calculated not only for spectral features but also for fundamental frequency and voicing strength features.

The objective is that the sCost model helps to discard units far beyond the average acoustics in the corpus and thereby contribute to select better quality units for concatenation. Additionally, since the HMM-based voice we use to generate parameters is trained with neutral style data, we expect that the sCost will penalise those segments (units) pronounced with a very different style. In some way, this approach is similar to the one described in [7], where synthetic speech data annotated as natural and unnatural is used to train a SVM model that helps to evaluate the naturalness of synthetic speech.

The paper is organised as follows. In Section 2 the methodology of sCost computation and its utilisation in unit selection synthesis is described. In Section 3 we describe how the neutral style HMM-based voice is created, the sCost model is calculated and how it is used in run time unit selection. In Section 4 the method presented in this paper is evaluated in a listening test, where a baseline unit selection voice is compared with two unit selection voices created with sCost model; main effects are discussed. Finally in Section 5, conclusions are made and future work is envisaged.

2. Methodology

The proposed methodology describes the usage of the HMM-based statistical model cost (i.e. sCost) in unit selection speech synthesis. We describe the procedure for estimating sCost from different parameters, using Dynamic Time Warping (DTW), and its use in selecting candidate units for synthesis.

2.1. Computation of sCost

As shown in Figure 1 the sCost is computed in several steps. As a first step, an automatic labeller estimates automatic segment labels based on recorded speech and phonetic transcription from text prompts. Secondly, an HMM voice is created by the HMM voice-building module using the automatic labels, generated in the previous step, and recorded waveforms. In the next steps, the HMM parameter generation module gen-

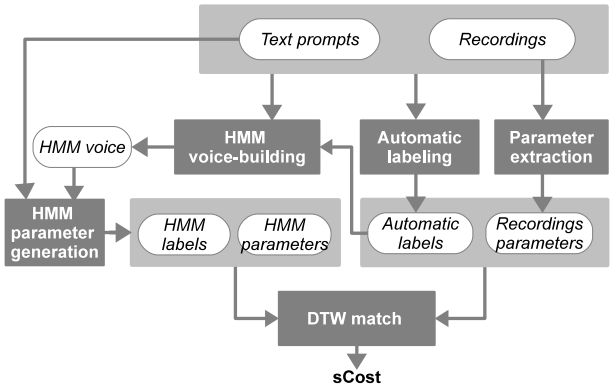


Figure 1: sCost computing methodology ([6])

erates parameters and HMM predicted segment labels from the text prompts. Having similar conditions for parameters dimension, frame size and frame-shift, parameters are extracted from the recorded waveforms. Finally, DTW computes an sCost by matching the extracted parameter feature vector sequence of the recorded speech and the generated parameters by the HMM parameter generation module. When aligning the two parameter vector sequences their corresponding unit segment labels are taken into account.

2.2. Unit selection using sCost

The unit selection based approach is based on: the selection of appropriate candidate units, which are close to the intended *target*, from a database of natural speech; and an appropriate combination of the selected units in order to achieve good speech quality. The unit selection algorithm plays a key role in identifying which of the available candidate units are appropriate for the target of intended speech to be synthesised.

According to the traditional unit selection algorithm [8], the algorithm includes two types of costs: *target cost* to define how well a candidate unit from the database matches the target unit; and *concatenation cost* to define how well two selected units can combine at joints. The cost functions can be written as the following:

$$targetCost(u_i) =< w, c(u_i) > \quad (1)$$

$$joinCost(u_i, u_{i-1}) =< w, c(u_i, u_{i-1}) > \quad (2)$$

where u_i is the candidate unit i; c is the cost vector containing several feature costs; and w is the weight vector for the features.

In the proposed method, each candidate is associated with a precomputed sCost (i.e. quality measure) for each parameter. The parameter specific sCost measures can be combined as the following:

$$sCost(u_i) = W^{\mathrm{T}} * \begin{pmatrix} sCostPAR_1(u_i) \\ ... \\ sCostPAR_n(u_i) \end{pmatrix} \quad (3)$$

where W is a weight vector; sCostPAR represents a parameter specific sCost measure.

The overall cost for selecting units in the dynamic programming stage can be modified as the following:

$$totalCost(u_i) = W_1^{\mathrm{T}} * \begin{pmatrix} targetCost(u_i) \\ joinCost(u_i, u_{i-1}) \\ sCost(u_i) \end{pmatrix} \quad (4)$$

At the stage of selecting units, the dynamic programming algorithm finds the best suitable candidates for the target by minimising the total cost function described above. Beam search is used to minimise the speed of computation.

3. Realisation

In order to test the method proposed in this paper several unit selection voices were created using the MARY TTS voice building tools [9]. One HMM-based voice and three unit selection voices were created using audio book data, in this case "Mansfield Park" released in the Blizzard Challenge 2013 [1]. The audio book data was already split into prosodic phrase level chunks. The sentence segmentation and orthographic text alignment of the audio book has been performed using an automatic sentence alignment method – LightlySupervised – as described in [10].

3.1. HMM-based voice building

HMM-based voices are well known to produce flat spectral trajectories and smooth F0 contours, which for our purposes will be a good approximation of the context-dependent average segment acoustics. Additionally, and in order to generate a HMM-based voice with a stable, not so expressive narrative style, we have used the same techniques used in [11] to create a neutral voice out of audio book data. That is, we have extracted acoustic features from each sentence of the corpus and perform principal component analysis so to discard sentences beyond a PC1 threshold. For this experiment we have extracted the following acoustic features:

- Fundamental frequency (F0) and F0 statistics: mean, max., min., and range.
- Number of words.
- Average energy, calculated as the short term energy averaged by the duration of the sentence in seconds.
- Voicing rate calculated as the number of voiced frames per time unit.
- Five band pass voicing strengths estimated with peak normalised cross correlation of the input signal.

For calculating voicing strengths, the input signal is filtered into five frequency bands and mean statistics of these measures are extracted per sentence. Voicing strengths features are normally extracted in the MARY TTS voice building framework for HMM-based synthesis using mixed excitation.

As in [11], we have found that also in this data, voicing rate and voicing strengths contribute more than F0 or MFC to the variance of the first principal component. This might indicate that the data contains more variation in speaking styles (voice quality) than extreme emotions. Using this method we have selected 3363 sentences out of the aprox. 7000 sentences of the whole audio book corpus, for building a HMM-based neutral voice. When creating a HMM-based voice in the MARY TTS framework, three types of acoustic features are extracted:

- MFC: Mel generalised cepstrum, dimension 25, extracted using SPTK [12],

- LF0: Log fundamental frequency, dimension 1, extracted using snack [13],
- STR: Voicing strengths, dimension 5, from 5 bands of frequency, extracted using snack and a set of filters provided in the MARY TTS voice building framework.

For the experiments with sCost model, these features were also extracted from the whole corpus, with which we created three unit selection voices, two of them employing sCost, as explained below.

3.2. sCost estimation for audio book data

DTW, a dynamic programing technique with optimal alignment to match the acoustically most similar sections between two phonetic segments, is implemented in MARY TTS for estimation of sCost between extracted parameters from recordings and generated parameters from the HMM voice. Here, an automatically labelled phone segment in the recorded speech is matched with the corresponding segment generated by the HMMs. The criterion for finding the optimal path is the Mahalanobis distance between the recorded and generated parameter vectors (i.e. MFC, STR, LF0), using the variance computed per phone on the recorded waveforms. sCost is computed as the sum of the Mahalanobis distance over the optimal path, divided by the number of frames in the recorded segment and in the generated segment.

MARY TTS unit selection uses diphones as basic units. An average of two half-phone sCost measures are considered as the diphone's sCost. In order to estimate sCost for each half-phone, the acoustic parameters are also extracted from the whole corpus of aprox. 7000 sentences, and generated using the HMM parameter generation component of the neutral HMM-based voice. In this work, we compute three sCost measures for each unit. They are: sCostMFC using MFC parameters; sCostSTR using STR parameters; sCostLF0 using LF0 parameters.

3.3. Unit selection voice building

The unit selection voice building use the standard approach in MARY TTS framework [14]. The only difference in the new voices is that they contain precomputed sCost measures in timeline files. All the precomputed measures are put into a timeline file, together with other timeline files in the unit selection voice.

As mentioned before, for testing the method presented in this paper we have created three unit selection voices, using the whole corpus (aprox. 7000 sentences), with the following characteristics:

- voice A: baseline voice, it does not use sCost model,
- voice B: a unit selection voice that uses a sCost model calculated with only MFC features, as in [6],
- voice C: a unit selection voice that uses a sCost model calculated with MFC, STR and LF0 features.

For run time synthesis, the MARY TTS unit selection algorithm combines the usual steps of pre-selecting candidate units, a dynamic programming phase combining weighted join costs and target costs, and a concatenation phase joining the selected units into an output audio stream. In the current version, a very small pre-selection tree is manually specified and can pre-select units, e.g., by their phone or diphone identity [14]. A beam search is used in the dynamic programming step to keep processing time low.

In addition to join costs and target costs, we add statistical model costs to the phase of dynamic programming phase as described in Eq. 4. Total sCost in this equation is measured with weighted sum of parameter-specific sCosts as the following:

$$sCost(u_i) = W_0^{\mathrm{T}} * \begin{pmatrix} sCostMFC(u_i) \\ sCostSTR(u_i) \\ sCostLF0(u_i) \end{pmatrix} \quad (5)$$

The weights of join cost, target cost and statistical model cost are tuned manually, "heuristically", for each voice based on subjective perception. For Voice A (no sCost), the weights for sCost becomes zero. For Voice B (sCost MFCs), the weights of sCostSTR and sCostLF0 becomes zero. To make a fair comparison, we manually tuned weights of all three voices to their best performance by listening to the synthetic speech of several random sentences.

4. Evaluation

Since audio book data is more expressive, it is difficult to define an objective measure, like spectral distance, to compare sentences that can be correctly pronounced in different ways. So in order to evaluate the effect of sCost we have performed a preference perceptual test, where we ask users to listen and compare pairs of sentences and select the one that in their opinion sounds better in quality and pronunciation for the given text. As test sentences we have selected 12 sentences from another book: "The adventures of Tom Sawyer", for reference we have included them in Figure 1.

As an example of the effect of sCost on the generated sentences, we can see in Figure 2 the F0 contour obtained with the three unit selection voices A, B and C for sentence 6. `Tom, what on earth ails that cat?`. We can observe that in this example the F0 contour generated with voice C is much more smooth than the contours generated with voices A and B. Perceptually, the sentence generated with voices A and B present much more variations in pronunciation, but with introduction of artifacts that degrade the quality. The sentence generated with voice C, on the other hand present a more stable narrative style, with better quality.

A more detailed, spectral view of the word `ails` in the same sentence, is presented in Figure 3. In this figure we can observe how the sCost model in the sentence generated with voice C, present a considerable reduction in spectral discontinuities. These observations seem to correlate with the results obtained in the listening test.

4.1. Listening test

Seventeen people participated in the listening test, among them several speech experts and most of them non-native speakers of English. The users listened in random order 12 pairs of sentences, in three sessions: AB, AC and BC. Where AB means that users listened the 12 sentences generated by unit selection synthesisers A and B.

As shown in Figure 4, the results indicate that broadly, subjects preferences are:

- voice A (72%) over voice B (28%),
- voice C (78%) over voice B (22%),
- voice C (58%) over voice A (42%).

Although overall preferences for voice A and C are high, subjects clearly indicate their preference towards 8 samples of

1. Well I WILL, if you fool with me.
2. Tom knew that when his name was pronounced in full, it meant trouble.
3. Huckleberry Finn was there, with his dead cat.
4. It was on a hill, about a mile and a half from the village.
5. The boys clasped each other suddenly, in an agony of fright.
6. Tom, what on earth ails that cat?
7. Some people think they're mighty smart, -- always showing off!
8. They had a famous fried-egg feast that night, and another on Friday morning.
9. I want to go home.
10. The stillness continued; the master searched face after face for signs of guilt.
11. Becky's face paled, but she thought she could.
12. The village was illuminated; nobody went to bed again; it was the greatest night the little town had ever seen.

Table 1: Test sentences used in in the listening test.

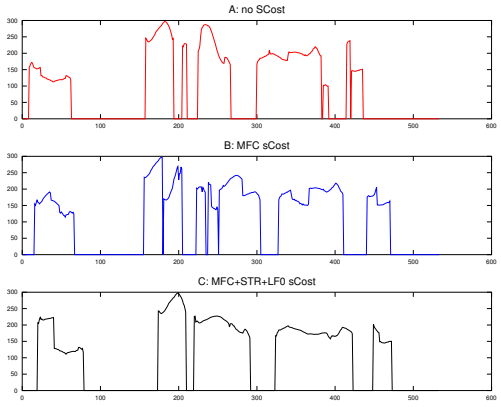


Figure 2: F0 contours of test sentence 6. "Tom, what on earth ails that cat?" generated with unit selection voices A, B and C.

voice C, 3 samples of voice A, and one sample for which they have almost equal preference.

4.2. Discussion

Among the synthesised examples[1], sentence 6 with text `Tom, what on earth ails that cat?` also reveals some interesting insights of the approach. The synthesised audio from system A is realised as `Tom, what earth ails that cat?` (deletion of "on"), whereas it is realised by system B as `Tom, what but on earth ails that cat?` (insertion of "but"). These errors are mostly due to misalignment in automatic labelling. However, such problems were successfully avoided by the realisation in system C. This means that sCost computed with all parameters, seems to deal with automatic labelling errors appropriately.

The average consecutive length (ACL) of each unit selection system are:

$$\begin{aligned} ACL_A &= 6.2 \\ ACL_B &= 3.1 \\ ACL_C &= 5.0 \end{aligned}$$

while the average consecutive length of units in system A is much higher than in systems B and C, it is much lower for system B. This means that the insertion of sCost into the unit selection algorithm reduces the average number of units that are consecutive, specially when sCost is precomputed for MFC only. In the listening test, reduction of ACL have had a negative effect on the performance of system B, because more dissimilar joins introduced more perceptible artifacts, that reduced the speech quality.

When the sCost uses all parameters including MFC, STR and LF0, the average consecutive length in unit selection is increased. Interestingly, the subjective preference is higher for system C when compared to system A, though the average consecutive length is lower. Therefore, it seems that system C is maintaining a fair balance between the consecutive selection of units and acoustically similar units.

A counter example is the following: in the synthesised sentence 9, `I want to go home`, the subjects fully preferred system A over system B and system C over system B. However, 75% of subjects preferred system A instead of system C. The average consecutive length of this particular sentence synthesised by systems A, B and C are 8.67, 2.89 and 5.2 respectively. The choice of voice A in this particular case might be due to less number of joins in the synthesised audio. Thus, we can conclude that, although sCost helps to reduce concatenation errors and make the voice style more stable, these type of errors still appear, so the approach will be further investigated in order to improve the join model in combination with sCost.

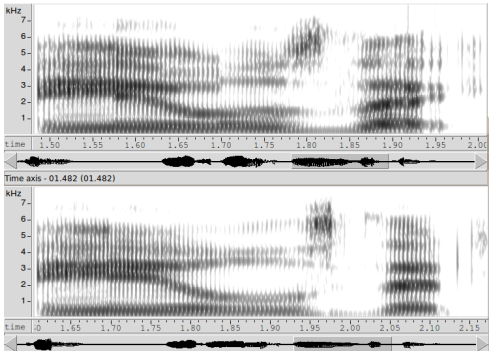


Figure 3: Spectrum of a section of sentence 6. corresponding to the word "ails" generated with unit selection voices A (upper spectrum) and C (lower spectrum).

5. Conclusions

In this paper we have presented the implementation and evaluation of a unit selection text-to-speech system that incorporates

[1] http://www.dfki.de/~charfuel/listening_test/listening_test.html

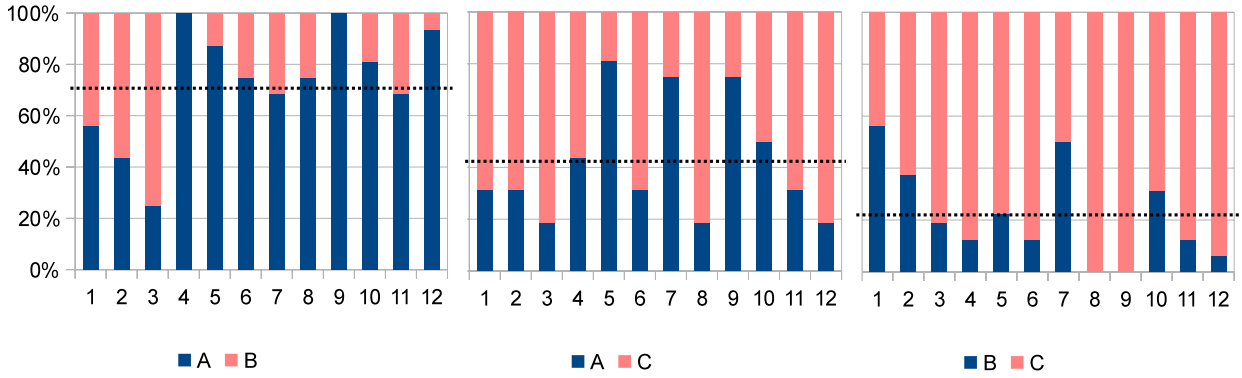


Figure 4: Listening test results: preference for 12 sentences synthesised with systems A and B, A and C and B and C. Dashed line in figures indicate the average preference between systems.

a statistical model cost (developed in a previous work [6]), in addition to target and join costs, for controlling the selection of unit candidates.

We have extend our previous work in two ways: (i) sCost is used in addition to target and join costs for controlling the selection of unit candidates in a unit selection synthesiser; and (ii) sCost is calculated not only for spectral features but also for fundamental frequency and voicing strength features. The method has been tested on unit selection voices created using audio book data. Due to the highly variable expressivity of the data, the HMM-based voice used to calculate sCost was built with neutral style data, automatically selected from the corpus.

Three unit selection voices were created, using all the data in the audio book, to perform a listening test where a baseline system without sCost was compared against: a system using a MFCsCost; and another using MFCsCost, STRsCost and LF0sCost. The listening test results indicate a clear preference for the system that include the three types of sCost. We have also discussed and presented examples of the effect of sCost on the F0 contour and spectrum, as well as, the effect on the average consecutive length of units.

We have shown how the use of sCost based only on spectrum introduce more variety on style pronunciation but affects quality; whereas using sCost based on spectrum, pitch and voicing strengths improves significantly the quality, maintaining a more stable narrative style. In future work we will not only investigate a better join model that suits for this approach, but also work towards a generic approach for style control using the proposed statistical model cost measures.

6. Acknowledgements

This work is supported by the European Union Seventh Framework Programme under grant agreement n288241 through the Michelangelo project. This work is also supported by the EU project SSPNet (FP7/2007-2013) and partially supported by AVATAR 1.1 project.

7. References

[1] S. King and V. Karaiskos, "Blizzard Challenge 2013," http://www.synsig.org/index.php/Blizzard_Challenge_2013.

[2] Z.-H. Ling and R.-H. Wang, "HMM-based hierarchical unit selection combining kullback-leibler divergence with likelihood criterion," in *Acoustics, Speech and Signal Processing, 2007. ICASSP 2007. IEEE International Conference on*, vol. 4, 2007, pp. IV–1245–IV–1248.

[3] Z.-H. Ling, L. Qin, H. Lu, Y. Gao, L.-R. Dai, R.-H. Wang, Y. Jiang, Z.-W. Zhao, J.-H. Yang, J. Chen, and G.-P. Hu, "The USTC and iFlytek speech synthesis systems for blizzard challenge 2007," in *Proceedings of Blizzard Challenge 2007*, Bonn, Germany, 2007.

[4] A. W. Black, C. L. Bennett, B. C. Blanchard, J. Kominek, B. Langner, K. Prahallad, and A. Toth, "Cmu blizzard 2007: A hybrid acoustic unit selection system from statistically predicted parameters," in *Proceedings of Blizzard Challenge 2007*, Bonn, Germany, 2007.

[5] Y. Qian, Z.-J. Yan, Y.-J. Wu, F. K. Soong, G. Zhang, and L. Wang, "An HMM trajectory tiling (HTT) approach to high quality TTS – microsoft entry to blizzard challenge 2010," in *Proceedings of Blizzard Challenge 2010*, Kansai Science City, Japan, 2010.

[6] S. Pammi, M. Charfuelan, and M. Schroder, "Quality control of automatic labelling using HMM-based synthesis," in *Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on*, 2009, pp. 4277–4280.

[7] H. Lu, Z.-H. Ling, L.-R. Dai, and R.-H. Wang, "Building HMM based unit-selection speech synthesis system using synthetic speech naturalness evaluation score," in *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*, 2011, pp. 5352–5355.

[8] A. J. Hunt and A. W. Black, "Unit selection in a concatenative speech synthesis system using a large speech database," in *Acoustics, Speech, and Signal Processing, 1996. ICASSP-96. Conference Proceedings., 1996 IEEE International Conference on*, vol. 1. IEEE, 1996, pp. 373–376.

[9] MARY TTS, "VoiceImportTools Tutorial," https://github.com/marytts/marytts/wiki/VoiceImportToolsTutorial, 2012.

[10] N. Braunschweiler, M. Gales, and S. Buchholz, "Lightly supervised recognition for automatic alignment of large coherent speech recordings," in *Interspeech*, Makuhari, Chiba, Japan, 2010.

[11] M. Charfuelan and I. Steiner, "Expressive speech synthesis in MARY TTS using audiobook data and EmotionML," in *Proceedings of Interspeech 2013*, Lyon, France, 2013.

[12] S. Imai, T. Kobayashi, K. Tokuda, T. Masuko, K. Koishida, S. Sako, and H. Zen, "Speech signal processing toolkit (SPTK), Version 3.3," http://sp-tk.sourceforge.net, 2009.

[13] K. Sjölander, "The snack sound toolkit," http://www.speech.kth.se/snack, 2012.

[14] M. Schröder, S. Pammi, and O. Türk, "Multilingual mary tts participation in the blizzard challenge 2009," in *Proc. Blizzard Challenge*, vol. 9, 2009.

Understanding Factors in Emotion Perception

Lakshmi Saheer, Blaise Potard

Idiap Research Institute, Martigny

lsaheer@idiap.ch, bpotard@idiap.ch

Abstract

Emotion in speech is an important and challenging research area. Addition or understanding of emotions from speech is challenging. But, an equally difficult task is to identify the intended emotion from an audio or speech. Understanding emotions is important not only in itself as a research area, but also, for adding emotions to synthesized speech. Evaluating synthesized speech with emotions can be simplified if the correct factors in emotion perception can be first identified. To this end, this work explores various factors that could influence the perception of emotions. These factors include semantic information of the text, contextual information, language understanding and knowledge. This work also investigates the right framework for a subjective perceptual evaluation by providing different options to the listeners and checking which ones among these are the most effective response to evaluate the perception of the emotion.

Index Terms: Emotion perception, perceptual factors, emotions in speech, emotional speech analysis/synthesis, metrics for subjective perceptual evaluations

1. Introduction

In recent years, many researchers have been studying the area of emotional speech, how to synthesize, recognize or even interpret emotions from human speech. Emotion in speech is a rather complex topic compared to other research areas in speech, particularly due to the absence of a standard metric to gauge the emotional content of a speech sample.

By contrast, automatic speech recognition (ASR) can be evaluated in a simple manner using objective scores such as word or phone error rates. Text-to-speech synthesis, while harder to evaluate parametrically, still has objective and subjective scores related to quality or intelligibility. There are no such parameters to classify the emotions expressed in speech. Listeners usually identify the emotions subjectively, according to their mood, opinions and cultural background. As a result, a given sample of emotional speech could be perceived differently by each listener.

There are various techniques used to generate the emotion in synthesized speech. A short review of the different emotion generation techniques was presented in [1]. There are 3 major techniques explained in [1] for addition of emotions: (1) Formant Synthesis: the acoustic speech data is entirely generated using the rules on acoustic correlates of various speech sounds. Emotional expressivity is modeled by manipulating the parameters related to voice source and vocal tract. (2) Di-phone synthesis: the monotonous human recordings are split into di-phones and then concatenated during synthesis. The required F0 contour is generated through signal processing techniques, which along with some manipulation of the duration can generate the emotion expression to some extent. (3) Unit selection synthesis: units of variable size are selected from a large inventory of speech database which best approximates a desired target utterance defined by a number of parameters. Databases for different emotions are collected separately and corresponding units are concatenated for generating emotional speech. Apart from the techniques mentioned above, there are also efforts with parametric speech synthesis [2] to generate different speaking styles and expressions. It is easier to generate different emotions and speaking styles with a hidden Markov model (HMM) based parametric speech synthesis system by using a style vector as in [3]. A comparison of unit selection and HMM based emotional speech synthesis [4] revealed that unit selection methods require improvements to prosodic modeling and that HMM-based methods require improvements to spectral modeling for emotional speech synthesis and that certain emotions cannot be reproduced well by either method.

The accuracy of humans listeners for the emotion identification task for recorded speech databases is approximately 80% [5]. This number will further degrade if a machine generated emotion or speech is being evaluated. The degrees of perceived emotions also varies from speaker to speaker and across listeners. Various techniques have been proposed to evaluate emotion - both objectively and subjectively [6]. The objective scores include classification of speech based on spectral or prosodic features for emotion identification using machine learning algorithms. Subjective tests are more popular with emotion evaluation since emotions are supposed to be perceived more realistically by human listeners. The most popular subjective tests include forced categorization of emotions into prespecified classes, or a descriptive free response system. The evaluation task can be made more intricate by assessing the degrees of naturalness, believability or overall preference of the emotion expression (often on a five-point scale) in addition to the forced categorization [7].

There are different perspectives to interpret emotions, in particular a psychological and a signal processing perspectives. The psychological perspective consists of studying different states or degrees [8] (e.g. activation, valence, dominance) to differentiate emotions. The signal processing perspective studies the changes in the speech signal attributes corresponding to different emotions [9]. There are also different ways of evaluating the emotion perception as summarized in [1] which represents the "perceptual" view for understanding emotions. This study aims at collecting these different streams of emotional research under one roof with the aim of studying how the different emotion generation techniques can be effectively evaluated in future. Towards this end, this work performs perceptual evaluations for different emotions with/without corresponding semantic and contextual information. The influence of language knowledge and understanding is also evaluated for the perception of different emotions. Finally, the best mode of response from listeners is evaluated among the three response types pro-

posed: emotion classification using a forced choice task, emotion identification as a free text response and identifying the degrees/states for different emotions.

This paper is organized as follows. Section 2 briefly describes the factors that could influence the perception of emotion, followed by the design of the subjective listening tests explained in section 3. Finally, the deductions from the evaluations are summarized in section 4.

2. Factors affecting emotion evaluation

There are various factors that affect the perception of emotion, including the mood and perception of the listeners. Studying an exhaustive list of all such factors would greatly exceed the scope of this paper. In this work, we focus on four basic factors that could influence the emotion perception: (1) Parameters of the speech signals that indicate the particular emotion; the basic features include pitch, duration and intensity, but there are numerous other voice quality features as well. (2) Semantic content of the text that is being spoken; we are interested in studying whether people can perceive the emotion if the semantic content of the text does not match the emotion that is being expressed. This could in turn lead to the topic of representing irony[1] in synthesized speech. (3) Context Information which is intended to help understand the mood or situation of the speaker. (4) The language ability of the listener could influence the perception of emotions, as it directly influences the understanding the semantics of the emotional speech samples.

This work hypothesizes that these four factors influence the perception of emotion, and the authors expect the results of the evaluations to reveal whether the hypothesis can be proved. Since it is not possible to gauge mood of the listeners or the variation in emotion perception by different listeners, these factors are usually (assumed to be) statistically normalized using a large enough number of listeners (of the order > 10). The different factors evaluated are detailed below.

2.1. Speech parameters

The emotional speech researchers have always relied on specific speech parameters that vary when representing a particular emotion. These features include pitch, duration, intensity, as well as some other voice quality features [10]. These features could be manipulated to modify the emotion in speech; the same features could be analyzed to perceive the intended emotion, which could lead to the generation of objective scores for emotion recognition. It is not clear however if human listeners use these cues in speech to perceive emotion. In order to understand if these cues play an important role in emotion perception, this work evaluates original recordings of emotional speech acted out by human actors. Compared to the evaluation of synthetic signals, this eliminates any direct effect from the distortions introduced by speech synthesis technology or emotion manipulation techniques. This way, the listener is not evaluating any particular emotion generation technique, but the intended emotions as expressed by humans.

2.2. Semantic content

A spoken utterance has two main components, the speech signal and the text that is spoken in the speech. The semantic content of the text cannot be isolated from the speech perception [11]. This factor might have a significant influence in the perception of emotion. Even though the listeners are usually asked to ignore the meaning of the text represented by the speech, it is not easy to totally isolate this influence. Most of the evaluations try to use "emotionally neutral" text. Neutral text itself can be classified into two types: one that fits into multiple emotions depending on the context - for example, the phrase "I did not expect this", could be expressed as happy, sad, or angry depending on the context in which it is spoken; the second type consists of neutral text that does not represent any particular emotion - for example, the phrase, "Whales live in the sea" does not convey any specific emotion. By opposition, emotive text is clearly intended to convey a specific emotion; an example of an emotive sentence is "I won a big lottery" and is clearly an "excited" or "happy" sentence. The emotions can be evaluated using sentences that represent neutral or emotive text (corresponding to the emotion). Similar studies were performed earlier with synthesized emotional speech [9]. This study further emphasizes the influence of the meaning of a sentence in perceiving the intended emotion.

2.2.1. Irony Effect

The influence of meaning can be further developed into the study of what is called the "irony effect". The irony effect refers to the situation where the speech express the opposite of (or something other than) the literal meaning of the sentence [12]. Other studies[19] attempt to synthesize the irony effect in machine-generated speech, by using an emotion different from the intended emotion in the text representing the speech. In this study, the irony effect is evaluated by using emotional speech for emotive sentences with different emotions, including ones that are opposite to the intended emotion in the text.

2.3. Context information

Most of the studies on emotional speech is based on a single sentence or phrase. Listeners are asked to evaluate the emotion on the basis of the perception from this sentence. This is a difficult task if the intended text is neutral or does not represent the whole mood of the speaker. The context in which the speech was spoken is particularly important for a listener to perceive the emotion of the speaker and the intended emotion in the sentence. This is similar to a real life scenario where the emotions are in a continuum and a particular emotion is expressed strongly in very rare situations. People usually understand the emotion of the speaker from the whole context and not from listening to ad-hoc sampled sentences. This study presents the listener with a detailed context in which the sentence is spoken and checks if the intended emotions are correctly identified in this case. Similar methodology was also tried in [9]. This method will recreate the real life scenario of how emotions are perceived by humans. The context information can be provided as an audiovisual content or even as a dialogue system. In this work, it is presented as a machine-generated background audio story with a few dialogues, which is efficient enough to convey the sentence context.

2.4. Language ability

A major factor affecting the emotion perception is the cultural impact. Different cultures represent emotions differently and the intended emotion might be totally different in different cultures even with the same speech parameters. There has been some study in this area [13]. An important factor to consider

[1] generally defined as *an expression or utterance marked by a deliberate contrast between apparent and intended meaning*[18]

is that the language ability might not only bias the listener towards the semantic meaning of the text, but also influences the emotion perception based on the cultural understanding of the language. To this end, emotional speech in a foreign language (German) is evaluated with some listeners who understand and others who have no knowledge of the language. In order to restrict the length of the evaluation, only one foreign language apart from English is evaluated.

3. Experimental Design

Recorded emotions of human actors are evaluated in this work, by opposition to machine-generated speech, so as not be subjected to the undesirable artifacts generally associated with synthesized emotional speech. This section gives the details of the design of the subjective evaluations performed to study the perception of the emotions based on the factors mentioned in the earlier section. The test is divided into four sections to evaluate the four factors mentioned above.

The first section represents the most commonly used experimental setup for emotion evaluation. The semantically neutral sentences are evaluated with different emotions in speech. The data used in this section comes from the EMA database [14]. Three semantically neutral sentences spoken by a male and a female speaker have been selected for the evaluation. Each sentence was spoken with four different emotions, namely happy, sad, angry and neutral, resulting in 24 sentences to be evaluated in this section.

The second section is similar to the first one but with speech from emotive text. Three different emotive sentences (with happy, sad and angry textual emotions) spoken by the same male and female speaker each with four different emotions (happy sad, angry and neutral) resulting in the evaluation of 24 sentences in this section. This section will help us study the influence of the semantic content of the speech signal. Since the same emotive sentence is used with different intended emotions in speech, this section may also give some insight on the perception of irony.

The third section represents the test for the influence of the contextual information in the perception of emotion. The same three emotive sentences used in the section 2 were presented within a context. Only the emotions corresponding to the emotive text was used for the test for both male and female speaker, which resulted in 6 sentences to be evaluated in this section. A state of the art commercial text-to-speech system[17] was used to generate the story around the emotional speech sample, to give a complete picture of the emotional state of the speaker. A voice very close to the speaker speaking in a neutral tonality was used to generate the context information, and the listeners were asked to ignore any distortion in the machine generated voice and evaluate the perception of the emotion from the single sentence highlighted as spoken by the human actor within the context of a neutral machine generated speech.

The last section is to study the influence of the knowledge of the language on the perception of emotion. The hypothesis to be tested here is that the knowledge of the language or the culture could improve the perception of the emotion. The tests were performed with emotional speech acted by a male speaker based on neutral text in German language from the German EmoDB [15]. Three sentences spoken with four different emotions including happy, sad, angry and neutral resulted in 12 sentences to be evaluated. Having listeners who do not understand any German may also allow us to study the perception of emotions without any semantic knowledge.

The responses of the listeners can be collected according to different response types as explained in [1]. The usual method of emotion evaluation is a discriminative task in which listeners are forced to select a particular emotion from a list of available emotion types. This, as mentioned in [1], is a discriminative task and not an emotion identification or descriptive task. This type of response makes the task simple and easy to evaluate. This task can be improved by adding a number of "distractor" response categories of emotions introduced in the perception test [9]. Other researchers [9] ask the listeners to describe in their own words what emotion they perceive. The listeners could provide a free response with keywords. These keywords are then grouped and classified into meaningful categories to identify the emotion perceived by the listener.

There are also ways to parameterize the emotions on different scales [8] based on states representing the emotions. This is similar to the scale represented by the FEELTRACE [16] concept representing emotions in a continuum between the space of valence and activation. This paper categorizes the emotional states as explained in [8] (also as conceptualized by the psychologists for the communication of affect), into three dimensions. These dimensions are usually termed as arousal, pleasure and power. Pairs of adjectives like happy/unhappy or pleased/annoyed (for pleasure), agitated/calm or excited/apathetic (for arousal) and powerful/powerless or dominant/submissive (for power) can be used to represent these three dimensions.

The types of responses in the evaluations for this work include a forced choice discriminative task with the four emotions actually used (happiness, sadness, neutral, anger) plus two distractor emotions (fear and surprise). Fear and surprise could be easily confused with sad and happy respectively. Also, the listeners are asked to provide a free response based on their perception of the emotion. A few listeners did not find this free response option useful and left the input space blank or mentioned "same as selection" or mentioned the same emotion they selected in the forced selection. This evaluation then presented three dimensions to evaluate or classify different emotions. These include valence, arousal/activation and dominance. Each of the three dimensions were varied on a five point Likert scale: Valence ranging from annoyed (negative) to pleased (positive), Activation or arousal ranging from very calm to very agitated/excited and Dominance varying from powerless (submissive) to powerful (dominant). The mid point of each dimension referred to as neutral may in turn represent a non-emotional speech. Apart from these responses, the "emotive text" section (section 2) also included a response to check if the emotional speech was perceived as irony or not. There were three choices: yes, no and maybe.

The test was available online, and was sent out to members of the research communities, mainly working on speech and signal processing. 16 listeners participated in the test, out of which 3 were native German speakers. The listeners were from different nationalities and cultures. Seven listeners did not understand any German while others varied from "can read/write" to "can understand bits". The test was rather long due to the multiple modes of responses and listeners found it a bit tiring especially the descriptive part of explaining the emotion in words.

4. Results and Discussions

Results of the perceptual experiments can be summarized with confusion matrices for the four sections separately. The confusion matrices are based on the forced choice test, which in-

Emotion	Keywords
Angry	quite excited, a bit annoyed, agitated, upset, disgust, panic, aggressive, pissed off, distressed, energetic, dominating, threatening, menace, stressed, disagreeing, jeering, unsatisfied, upset, declarative, impatient, strong, frustration, threat
Happy	elevated, enthusiastic, pleased, glad, ecstatic, excited, laughing, amused, having fun, optimistic, playful, joking, interested, content, optimistic, looking forward, confident, interested, positive emotion
Neutral	no emotion,disgusted, explanatory, declarative, determined, dubitative, unsure, calm, dubitative, questioning, tired, interested
Sad	upset, slightly disgust, grief, depressed, bored, confused, broken, beaten down, stoned, monotone, crying, tired, whispering, disappointed, weary, desperate, bored, empathic, melancholic, uninterested, nervous

Table 1: Keywords for emotion classes

cluded the two distractor emotions "fear" and "surprise". The descriptive response almost always had the same meaning as the forced choice representation and some listeners felt it as a redundant response that could be ignored. The keywords corresponding to the different classes are grouped together in Table 1. In a couple of isolated cases, listeners described the emotion correctly, but chose a different (wrong) emotion from the forced choice, but there were only very few cases that thus benefited from the descriptive response. This limited gain gets neutralized by other confusing keywords which do not have a clear class like the keyword "upset" which could represent either sad or angry. For these cases, the values given to the dimensions might help disambiguate the situation. To summarize, listeners preferred the forced choice due to convenience, and the descriptive response did not give any significant performance gain. The different degrees for the three dimensions were in good agreement across users.

The section evaluating the influence of the contextual information (section 3) lead to very accurate emotion classification. This task is however slightly confusing due to the presence of audio from two different origins in the stimuli: neutral machine-generated context speech, and emotional speech spoken by a human actor; the listeners were instructed to base their evaluations on the emotional speech. The listeners managed to correctly perform the evaluation task except for one listener who classified all data in this section as neutral, due to the neutral machine-generated contextual sentences. This listener has thus been omitted from the results of section 3. The results and corresponding observations for each section of the test are detailed below.

4.1. Section 1: Neutral Text

This section uses emotionally neutral text to evaluate the perception of emotion in speech. This is the classic way of evaluating emotional speech with the argument that when the text is semantically neutral listeners tend to focus on speech parameters conveying emotion. The results of the forced choice emotion discrimination is presented in Table 2. The table shows confusion matrix for the identified emotion with the underlying intended emotion in the speech. Angry and neutral have the best performance as observed in the literature, followed by the sad emotion. Happy emotion has the worst performance and is confused with surprise and neutral. The distractor emotion (surprise) here increases the confusion, degrading the performance. The table shows both recall and precision for representing the performance. All the performances are well above chance of 25%. The precision values are quite high even for the emotions that have a smaller recall. It is not clear whether listeners are ignoring the semantic meaning of the text and concentrating on the speech parameters since happy emotion is confused mainly with neutral, and neutral emotion has the best recall.

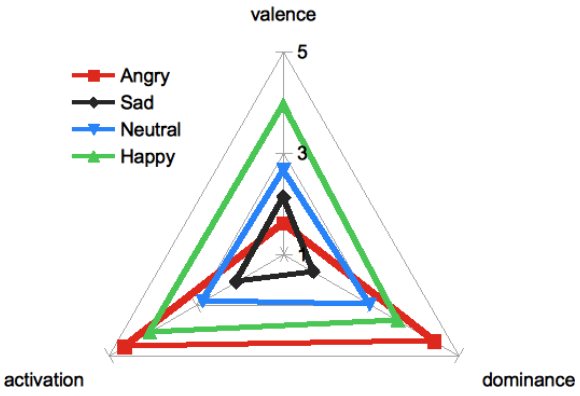


Figure 1: Average values for dimensions

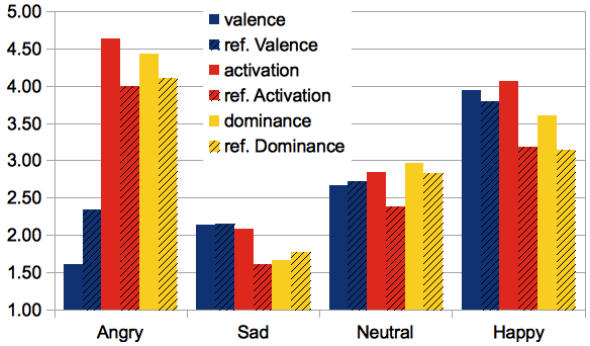


Figure 2: Comparison of Dimensions

In some cases, the neutral text with a particular emotion may be perceived emotive and hence, wrongly recognized (happy as surprise). This gives neutral emotion poor precision even with a good recall.

For the descriptive task, the keywords mostly correspond to the forced choice emotion selected. There are a few responses where the emotion is described correctly, but the forced choice emotion is wrongly selected. For example, selecting neutral in the forced choice task, but correctly describing as "tired or disappointment" for sad, "dominating" for angry, describing "happy" for happy (but selecting surprise) and "excited" for angry (selecting happy/surprise). Also, there are some responses that correctly identify the emotion in the forced choice task and describe them as a different emotion. For example, describing the speech as "annoyed", "depressed" or "angry" after correctly selecting the neutral and describing as "panic" after correctly choosing angry emotion in the discriminative task.

The different dimensions of the emotions are plotted in Figure 1 and also compared with what was reported with the database in Figure 2. The results of human evaluations performed by 18 listeners were distributed originally along with the database. Both results are comparable and follow the same trend across different emotions. The dimensions show clear regions of influence for each emotion with neutral emotion concentrating on the middle (at zero values). As seen in literature, the happy emotion overlaps with others in these dimensions and is not easy to differentiate.

4.2. Section 2: Emotive Text

The performance of the emotion identification improves for angry and sad with emotive text (as seen in Table 2), and performance of neutral and happy degrades. The degradation of neutral might indicate that the perception is based on the emo-

Sections	Neutral Text				Emotive Text				Context Info.		
Emotions	**A**	**S**	**N**	**H**	**A**	**S**	**N**	**H**	**A**	**S**	**H**
Angry	**75**	0	11	2	**90**	4	21	7	**29**	0	1
Surprise	8	0	0	23	1	0	0	25	0	0	2
Sad	1	**71**	7	5	1	**75**	16	0	0	**29**	0
Neutral	7	14	**77**	14	4	11	**59**	22	1	1	2
Happy	3	0	0	**51**	0	1	0	**41**	0	0	**25**
Fear	2	11	1	1	0	5	0	1	0	0	0
Recall	78.1%	74.0%	80.2%	53.1%	93.8%	78.1%	61.5%	42.7%	96.7%	96.7%	83.3%
Precision	85.2%	84.5%	68.8%	94.4%	73.8%	81.5%	61.5%	97.6%	96.7%	100%	100%

Table 2: Confusion matrix for forced choice test for first three sections

Dimensions	Valence				Activation				Dominance			
Emotions	**A**	**S**	**N**	**H**	**A**	**S**	**N**	**H**	**A**	**S**	**N**	**H**
Angry	-	0.004	0.004	0.002	-	0.002	0.002	0.007	-	0.002	0.002	0.004
Sad	0.004	-	0.007	0.002	0.002	-	0.004	0.002	0.002	-	0.002	0.002
Neutral	0.004	0.007	-	0.002	0.002	0.004	-	0.002	0.002	0.002	-	0.005
Happy	0.002	0.002	0.002	-	0.007	0.002	0.002	-	0.004	0.002	0.005	-

Table 3: p-values (2-tail) for the statistical significance test using Wilcoxon signed rank test over the average values of the 3 dimensions obtained for each sample, across emotions. All emotions have statistically different values for the three dimensions as the p-values are less than 0.01.

tive text (speech semantics). The happy emotion is confused with neutral and surprise. There is a lot of confusion when the text emotion and speech emotion have a mismatch, especially with the happy emotion in speech. The happy and sad emotions have a good precision even with poor recall. The Angry emotion has good recall but poor precision, indicating it is confused with others. Some of the descriptive responses relate to happy emotion class, but the forced choice is chosen as neutral. Most of the happy speech with mismatched emotive text is treated as irony.

The results for the three different dimensions are very similar to the results in the earlier section and also have the same correspondence to the human evaluation values distributed with the database. This indicates that the listeners were consistent with their feedback on different dimensions of emotions and the textual emotions did not appear to bias their response. The statistical significance for the emotions across the three dimensions are mentioned in the Table 3 based on the Wilcoxon signed rank test with a significance level of 0.01 for the combined results from Sections 1 and 2 (neutral and emotive text). The table shows that all systems have values that are significantly different from each other. People perceive the emotions as having different values across valence, activation and dominance.

There are three different types of text and speech emotion combinations. When both emotions match, the combination can be termed as a "matched" case. The combinations of Sad / Happy or Angry / Happy can be termed as "opposite" emotion pairs or "strong mismatch", all other combinations can be termed as "ambiguous" (or "mismatch"). The matched, ambiguous and opposite combinations are plotted in Figure 3. The "matched" modality does not lead to a statistically different perception of irony from the "ambiguous" one. However, the "strong mismatch" leads to a significantly stronger sense of irony than the other two conditions (based on Wilcoxon signed rank test, both 2-tail p-values less than 0.01). More precisely, "angry" sentences in a happy voice and "sad" sentences in a happy voice are perceived as the ones with the strongest potential for irony.

This task further emphasizes that listeners are biased by the

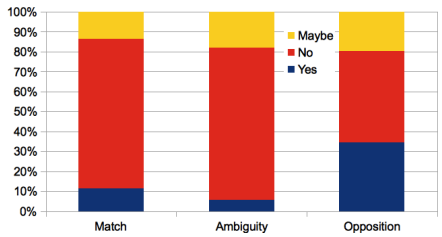


Figure 3: Results for Irony

semantic content of the text for the forced choice selection. The performance is best when the semantic content is matching with the intended emotion in the speech, and irony can be perceived easily when there is a mismatch in speech and text emotions. This supports the hypothesis that listeners tend to be influenced by the semantic content of the text when perceiving emotions. Contradicting this observation, even with opposing emotions in text and speech, some emotions like angry were almost always correctly identified. This also indicates that some emotions such as angry (with very prominent speech factors) are easy to identify even in adverse situations where the semantics in the text do not correspond to the emotion (even when perceiving it as irony).

4.3. Section 3: Context Information

This section provides more details to the listener on the background of the mood of the speaker, through the use of context information spoken in a neutral voice. This information appears to greatly improve the performance, as all emotions appear to be better recognized. As mentioned earlier, one listener did not understand the task and marked all tests as neutral, probably based on the contextual information presented using synthesized speech with a neutral voice. This listener is excluded from the results of this section. As hypothesized the context information has a big influence on the performance. Emotions "sad" and "happy" are significantly better discriminated in section 3, while "angry" is equivalent in both cases. All emotions have good precision with the context information compared to the section 2 without context information. The descriptive response corresponds exactly to the forced choice response. The

Native	German				Non-German			
Emotions	**A**	**S**	**N**	**H**	**A**	**S**	**N**	**H**
Angry	**7**	0	1	0	**35**	0	2	7
Surprise	0	0	0	2	1	0	0	6
Sad	0	**5**	0	0	0	**28**	0	0
Neutral	2	2	**8**	3	0	7	**34**	0
Happy	0	0	0	**4**	0	0	0	**23**
Fear	0	2	0	0	0	1	0	0
Recall	77.8%	55.6%	88.9%	44.4%	97.2%	77.8%	94.4%	63.9%
Precision	87.5%	100%	53.3%	100%	79.5%	100%	82.9%	100%

Table 4: Confusion matrix for forced choice test for Section 4 with listeners with and without German knowledge.

results for different dimensions in this section were also similar to the ones mentioned in the earlier section.

4.4. Section 4: Knowledge of Language

This is an interesting task where people with and without German knowledge were asked to evaluate the emotional speech in German. The text was semantically neutral. It can be seen from the forced choice results in Table 4, that native German speakers perform worse than non-native speakers. Both sad and happy have same precision for native and non-natives with better recall for non-natives. Neutral has better precision and angry has worse precision for non-native German listeners. The number of German speakers is however too small to make any generalizations. The results appear to contradict the hypothesis that the language or culture knowledge improves the emotion identification performance. It is possible the German listeners were biased because of the understanding of the semantic content of the text. The non-native speakers base their judgment only on the signal cues.

The native German speakers described the emotions more closely to the intended emotions, like, “Nervous” or “tired” for sad and “agitated” or “strong” for angry, even though choosing wrong emotion in the forced choice test. The selection of different dimensions were similar to the results in the earlier section.

5. Conclusions

All factors studied in this paper are shown to influence the perception of emotion, with the greatest emphasis on the contextual information and the type of emotion. Happy is usually a very difficult emotion to perceive. The semantics of the text has a great influence on the emotion perception. Though it may not be generalizable from this test, the current results suggests that if the intention is a pure evaluation of emotional content in the speech (comparing specific techniques for emotion generation), it might be better to use listeners without language knowledge to avoid any kind of bias from semantics. If the usability of the emotion in an application is to be checked, it is better to give a full background or context in which the emotional speech appears and listeners may be able to judge better. Forced choice task is the simplest method among the different techniques and does not deviate a lot from the descriptive response case. The happy emotion, which is difficult to perceive, is not discriminated well even in the three dimensions of valence, activation or dominance.

6. Acknowledgments

The work was supported by Eurostars Programme powered by Eureka and the European Community under the project “D-Box: A generic dialog box for multi-lingual conversational applications”.

7. References

[1] Marc Schröder, “Emotional Speech Synthesis: A Review”, Eurospeech, pp. 561-564, 2001.

[2] Heiga Zen and Keichi Tokuda and Alan Black, “Review: Statistical parametric speech synthesis”, Speech Communication, Volume 51(11): pp. 1039-1064, 2009.

[3] Takashi Nose, Junichi Yamagishi, and Takao Kobayashi. “A style control technique for HMM-based expressive speech synthesis”, IEICE Trans. Information and Systems, E90-D(9):1406-1413, 2007.

[4] R. Barra-Chicote, J. Yamagishi, S. King, J. Manuel Monero, and J. Macias-Guarasa, “Analysis of statistical parametric and unit-selection speech synthesis systems applied to emotional speech”, Speech Communication, 52(5):394-404, 2010.

[5] Dimitrios Ververidis and Constantine Kotropoulos, “Emotional speech recognition: Resources, features, and methods”, Speech Communication, Volume 48 (9), pp. 1162-1181, 2006.

[6] Alan W. Black et al., “New Parametrizations for Emotional Speech Synthesis”, Final Report for NPESS team - CSLP John Hopkins Summer Workshop 2011.

[7] Janet E. Cahn, “Generation of Affect in Synthesized Speech”, in Proc. of the Conference of the American Voice I/O Society. Newport Beach, California, 1989.

[8] Cécile Pereira, “Dimensions of emotional meaning in speech”, in Proc. of SpeechEmotion, pp. 25-28, 2000.

[9] Iain R. Murray and John L. Arnott, “Implementation and testing of a system for producing emotion-by-rule in synthetic speech”, Speech Communication, Volume 16 (4): pp. 369-390, 1995.

[10] C. Gobl and A. N. Chasaide, “The role of voice quality in communicating emotion, mood and attitude”, Speech Communication, Volume 40 (1-2), pp. 189 - 212, 2003.

[11] Gregory Hickok and David Poeppel, “Towards a functional neuroanatomy of speech perception”, Trends in cognitive sciences, Volume 4 (4), pp.131 - 138, 2000.

[12] David J. Amante, “The Theory of Ironic Speech Acts” Poetics Today, Volume 2 (2,) Narratology III: Narration and Perspective in Fiction (Winter, 1981), pp. 77-96.

[13] Klaus R. Scherer, “A cross-cultural investigation of emotion inferences from voice and speech: Implications for speech technology”, in Proc. of ICSLP, China, 2000.

[14] Sungbok Lee et.al., “An Articulatory study of emotional speech production.” Interspeech, Portugal, pp. 497-500, 2005.

[15] Felix Burkhardt et.al., “A Database of German Emotional Speech”, in Proc. of Interspeech, Portugal, 2005.

[16] Roddy Cowie et. al., “FEELTRACE’: An Instrument For Recording Perceived Emotion In Real Time”, in Proc. of ISCA Workshop on Speech and Emotion, pp. 19-24, Ireland, 2000.

[17] M. P. Aylett and C. J. Pidcock, “The cerevoice characterful speech synthesiser sdk”, in AISB, pp. 1748, 2007.

[18] American Heritage, “The American Heritage Dictionary of the English Language”, 4th ed. Houghton Mifflin Company, 2009.

[19] M. P. Aylett and B. Potard and C. J. Pidcock, “Expressive Speech Synthesis: Synthesising Ambiguity”, 8th SSW, Barcelona, 2013.

Multilingual Number Transcription for Text-to-Speech Conversion

R. San-Segundo[1], *J.M. Montero*[1], *M. Giurgiu*[2], *I. Muresan*[2], *S. King*[3]

[1]Speech Technology Group, ETSI Telecomunicación. UPM. Spain.
[2]Dept. of Telecommun., Tech. Univ. of Cluj-Napoca, Cluj-Napoca, Romania.
[3]Centre for Speech Technology Research, University of Edinburgh, UK.

lapiz@die.upm.es

Abstract

This paper describes the text normalization module of a text to speech fully-trainable conversion system and its application to number transcription. The main target is to generate a language independent text normalization module, based on data instead of on expert rules. This paper proposes a general architecture based on statistical machine translation techniques. This proposal is composed of three main modules: a tokenizer for splitting the text input into a token graph, a phrase-based translation module for token translation, and a post-processing module for removing some tokens. This architecture has been evaluated for number transcription in several languages: English, Spanish and Romanian. Number transcription is an important aspect in the text normalization problem.
Index Terms: Multilingual Number Transcription, text normalization, fully-trainable text conversion.

1. Introduction

Although Text to Speech (TTS) conversion is the area where more effort is devoted to text normalization, dealing with real text is a problem that also appears in other applications such as machine translation, topic detection and speech recognition when it is necessary to associate a phoneme sequence to a written word. In an ideal situation, there would be an unambiguous relationship between spelling and pronunciation. But in real text, there are non-standard words: numbers, digit sequences, acronyms, abbreviations, dates, etc. The main problem of a text normalization module consists of converting Non-Standard Words (NSWs) into regular words. This problem can be seen as a translation problem between a real text with NSWs and an ideal text where all the words are standard: unique relationship between word spelling and its pronunciation.

2. State of the art

One of the main references focused on text normalization is [1]. In this reference, authors propose a very complete taxonomy of NSWs considering 23 different classes grouped in three main types: numerical, alphabetical and miscellanea. Sproat et al describes the whole normalization problem of NSWs, proposing several solutions for some of the problems.

Additionally, it is also important to mention other references that have addressed specific problems included in the text normalization research line. Focused on abbreviations and acronyms, there are several efforts focused on extracting them from text automatically [2] and other efforts trying to model how they are generated [3]. Numbers [4] and proper names [5, 6] have been also the target of other research works. Number transcription has been missing from previous efforts and this paper contribute to complete this work [7]. Nowadays, much effort on text normalization is focused on SMS language interchanged through mobile phones and social networks like Facebook or Twitter [8, 9].

Due to the important advances obtained in machine translation in the last decade, there has been an increasing interest on using machine translation capabilities for dealing with the problem of text normalization [10, 11]. Text Normalization is an important aspect, not only for Text-to-Speech Conversion but also for Text Categorization [12] or Text Classification. [13].

3. Architecture description

Figure 1 shows the architecture diagram proposed in this paper. This architecture is composed of three main modules: a pre-processing module that splits the text input into a token graph, a phrase-based translation module based on Moses software, and a post-processing module for removing some tokens.

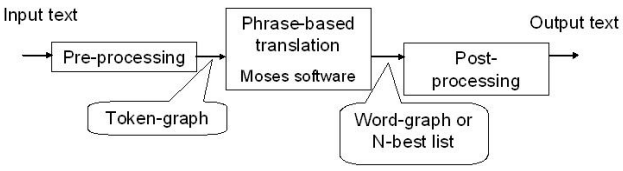


Figure 1: Architecture diagram

3.1. Pre-processing: Sentence Tokenization

In this first module, the input text is split into tokens. This process is carried out in two different steps. At the first step, a preliminary token sequence is generated considering a small set of rules. As one of the main targets of this work is to provide a language independent architecture, the main rules should be language independent:

- The first rule assumes that blank characters provide an initial segmentation in tokens.
- The second rule subdivides initial tokens considering some homogeneity criterions: tokens must have only alpha or numerical characters. If there is a change from alpha to number or vice-versa, the token must be subdivided. Secondly, punctuations characters are independent tokens.

Secondly, some of the tokens are re-written in a different format in order to facilitate their posterior translation. In this step, the idea is to classify each token as a standard word (W) or as a non-standard word (NSW). This classification can be done considering a dictionary of standard words in this language or considering a more complex classifier based on some features obtained from the target token and its context: character language model, vowels, capitals, etc. In this work, detecting numbers is quite simple based on the characters componing the token.

If the token is classified as a NSW, it is split into letters including some separators at the beginning and at the end of the letter sequence. For example, UPM (Universidad Politécnica de Madrid in Spanish) is rewritten into # U P M #. This way of rewriting an alpha token tries to introduce a high flexibility to facilitate the text normalization process. Considering sequences of letters, some unseen acronyms could be normalized by spelling (using the translations of its graphemes individually).

Also, all the numbers are rewritten dividing the token into digits. This work has considered two alternatives. In the first one, every digit is complemented with its position in the number sequence. For example: 2013 is rewritten as 2_4 0_3 1_2 3_1, where 2_4 means the digit 2 in the 4th position (position beginning from the right). The second alternative consists of dividing the number sequence in sets of three digits in sequence, complementing with its position in the sequence, and including additional tags to separate 3-digits sequences: 2013 is written as 2_1 tag1 0_3 1_2 3_1. The Roman numbers are first translated into Arabic ones and then, rewritten digit by digit. Ordinal numbers are not treated in this paper.

As it will be shown in the next section, the translation module can deal with graphs of token as input. Thanks to this possibility, it is possible to work with fuzzy decisions when classifying every token as standard word or NSW. Considering a token graph, both alternatives can be considered with different weight if necessary. Figure 2 shows an example of token graph for the sentence "Welcome to UPM2013"

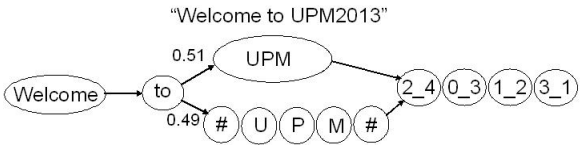


Figure 2: Token graph for the sentence "Welcome to UPM2013"

The token "UPM2013" is divided into two tokens: UPM and 2013. UPM, is rewritten considering two possibilities: as it is, and letter by letter. The second one is a number and it is rewritten digit by digit, considering the first alternative commented above.

3.2. Token Translation

The token translation is performed using a phrase-based system. The phrase-based translation system is based on the software released from NAACL Workshops on Statistical Machine Translation in 2012. The translation process uses a phrase-based translation model and a target language model. These models have been trained in accordance with these steps, see Figure 3.

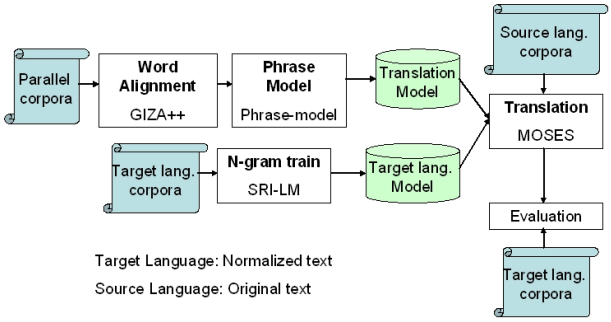


Figure 3: Process for training the translation and target language models

The first step is word alignment computation using the GIZA++ software [14]. In order to establish these alignments, GIZA++ combines the alignments in both directions. As there are many standard words, they are the same tokens in source and target languages, being important reference points for the alignment.

The second step is phrase extraction [15]. All token phrase pairs that are consistent with the token alignment are collected. Finally, the last step is phrase scoring. In this step, the translation probabilities are computed for all phrase pairs. Both translation probabilities are calculated: forward and backward.

The Moses decoder, available at http://www.statmt.org/moses/, is used for the trans-

lation process. This program is a beam search decoder for phrase-based statistical machine translation models. The N-gram language model has been generated with the SRI language modelling toolkit [16].

3.3. Post-processing

This module performs several actions in order to generate the normalized text to the speech synthesiser. One of the main actions is to remove unnecessary tokens. For example, if after the translation module there is any # token, used for defining the limits of the letter sequences, it must be removed.

Additionally, given that the translation module can generate a token graph or a sequence of N-best token sequence, it would be possible to add new translation modules in order to improve the translation process by considering new language models for reordering the N-best token sequences or searching the output token graph.

4. Multilingual Number Transcription

This section reports the experiments to adapt the text normalization module for multilanguage number transcription: English (EN), Spanish (ES) and Romanian (RO).

The database for the experiments has been generated randomly by taking into account several patterns of numbers including numbers with decimals. These numbers have been dividied into three sets: training (800), tuning (1000) or testing (5000). Every number belongs to only one of the sets. The evaluation measurements are BLEU, WER (Word Error Rate) and SER (Sentence Error Rate).

4.1. Number Tokenization

This section evaluates the two tokenization alternatives described above. In the first one, every digit is complemented with its position in the number sequence. For example: 2013 is rewritten as 2_4 0_3 1_2 3_1. The second alternative consists of dividing the number sequence in sets of three digits in sequence, complementing with its position in the sequence, and including additional tags between 3-digit sequences: 2013 is written as 2_1 tag1 0_3 1_2 3_1. In these experiments, the default values for the translation architecture have been considered: a 3-gram language model and grow-diag-final alignment. The sizes of the sets are 800 for training, 1000 for tuning and 5000 for testing.Table 1 shows improvement when considering the second tokenization alternative.

4.2. Token Translation

In the current experiments we have used in the training stage three types of alignments between the tokens in the source and the words in the target text with the aim to estimate statistical parameters for the number transcription system. Two types of word-to-word alignments ("tgt-tosrc" - target to source, and "srctotgt" - source to taget), as well as the "grow-diag-final" heuristic model, which consists of a number of intersections and unions aimed to cope with the asimetry of the word alignment models (Table 2). The best result has been obtained with different alignments depending on the language. Although the differences are not very high, the best alignment must be adapted depending on the language. The alignments are slightly dependent on the language due to the irregularities in number transcription and the use of language-specific function words. For example the number 30.000 is "thirty thousand" in English, but "treizeci de mii" in Romanian (functional preposition "de" is used), while 3.000 is "three thousand" in English, and "trei mii" in Romanian (without to use the preposition "de"). Therefore the heuristic alignment "grow-to-diag" would be more appropriate for Romanian. Analyzing the average number of words needed for transcribing the same number, we obtain 9.9 words per number for English, 10.1 words for Spanish and 13.3 words for Romanian, considering the same set of numbers with an average length of 8.3 digits, including decimals, dots and commas. For experiments in Tables 2 and 1, the three original sets were considered:

Tokenization			
EN	BLEU	WER	SER
1st alternative	97.9	1.6	7.4
2nd alternative	**98.5**	**0.8**	**6.8**
ES	BLEU	WER	SER
1st alternative	97.8	1.9	6.8
2nd alternative	**98.2**	**0.9**	**6.1**
RO	BLEU	WER	SER
1st alternative	98.5	0.9	5.4
2nd alternative	**99.2**	**0.5**	**4.6**

Table 1: Experiments with different tokenization

Aligment for training the translation model			
EN	BLEU	WER	SER
grow-diag-final	98.5	0.8	6.8
srctotgt	**99.4**	**0.4**	**4.4**
tgttosrc	98.1	0.7	6.4
ES	BLEU	WER	SER
grow-diag-final	98.2	0.9	6.1
srctotgt	98.2	0.9	6.1
tgttosrc	**98.5**	**0.7**	**4.5**
RO	BLEU	WER	SER
grow-diag-final	**99.2**	**0.5**	**4.6**
srctotgt	98.9	0.5	5.1
tgttosrc	98.5	0.8	7.8

Table 2: Experiments with different alignments used when training the translation model

training (800), tuning (1000) or testing (5000).

Training Set Size			
EN	BLEU	WER	SER
200 numbers	98.3	0.8	6.4
400 numbers	99.3	0.4	4.6
800 numbers	99.4	0.4	4.4
4000 numbers	**99.9**	**0.2**	**1.8**
ES	BLEU	WER	SER
200 numbers	97.3	1.5	8.8
400 numbers	98.2	0.9	6.0
800 numbers	98.5	0.7	4.5
4000 numbers	**99.6**	**0.2**	**1.0**
RO	BLEU	WER	SER
200 numbers	95.2	2.3	20.2
400 numbers	97.7	1.3	11.2
800 numbers	99.2	0.5	4.6
4000 numbers	**99.7**	**0.2**	**2.4**

Table 3: Experiments with different training sets

Finally, we have evaluated the influence of the size of the training set (Table 3). When decreasing the number of training examples the error increases. This increment is higher for those languages that need more words for transcribing a number. In these cases, we need more data to train the models. Analysing the errors, we have realized that many errors comes from the decimal part. In this case, the tokenization is not appropiate for this part. In order to analyze the influence of the decimal part, we carried out experiments with and without the decimal part considering only 200 numbers for training (Table 4). As it is shown, the error reduction is significant.

The influence of decimal part (200 numbers)			
EN	BLEU	WER	SER
With decimal part	98.3	0.8	6.4
Without decimal part	**99.3**	**0.4**	**3.6**
ES	BLEU	WER	SER
With decimal part	97.3	1.5	8.8
Without decimal part	**98.2**	**0.8**	**6.0**
RO	BLEU	WER	SER
With decimal part	95.2	2.3	20.2
Without decimal part	**97.9**	**1.1**	**11.2**

Table 4: Experiments without the decimal part using 200 numbers for training

4.3. Postprocessing

In this step, the main target is to avoid the presence of input tokens in the output sentence. If one initial token has not been translated, the postprocessing step replaces this token with the most probable translation: digits in our case. In this work, the postprocessing step did execute very few replacements, less than 0.1%.

5. Conclusions

This paper has presented a text normalization module to be integrated in a text to speech fully-trainable conversion system and its application to number transcription. The text normalization module proposed is based on statistical machine translation techniques. This module is composed of a tokenizer for splitting the text input into a token graph , a phrase-based translation module and a post-processing module for removing some tokens. This architecture has been evaluated for number transcription in English, Spanish and Romanian. For all the languages, the reached performance has been very good, specially for numbers not including decimals. When increasing the amount of data used for training the system, the results are better. Finally, it is necessary to comment that the system tuning, as the aligment of the token translator, must be adapted to the language in order to get the best results. Comparing to previous works, for example in [7], authors compare the language dependent (language specific) - rule based approach with the SMT and suggest to post-correct the results of LS-rule based by applying the SMT. This paper directly use the SMT, without any rule or language specific interventions. The system, at the end, only does minor post-corrections at a very small amount of data (eg. 0.1%). In [7], for a larger training dataset (eg. 3000 sentences) they obtain a BLEU=94,4, while in these experiments, for the smallest training set of 200 sentences, the BLEU is 95.2 (RO), 97.3 (ES) and 98.3 (EN). In [9], for SMS and Twitter messages, the BLEU is 99,2 for a larger training set, 90.000 sentences.

6. Acknowledgements

The work leading to these results has received funding from the European Union under grant agreement (n 287678). It has also been supported by TIMPANO (TIN2011-28169-C05-03), INAPRA (MICINN, DPI2010-21247-C02-02) and MA2VICMR (CAM, S2009/TIC-1542) projects. Authors also thank the other members of Simple4All project for the discussion on these topics.

7. References

[1] Sproat, R., Black, A., Chen, S., Kumar, S., Ostendorf, M., and Richards, C., 2001. *A Normalization of non-standard words*. Computer Speech and Language, 15(3), 287-333, 2001.

[2] Chang, J.T., Schtze, H., and Altman. R.B., 2002. *Creating an Online Dictionary of Abbreviations from MEDLINE. JAMIA*.

[3] Pennell D., and Liu Y., 2011a. *Toward text message normalization: Modeling abbreviation generation*. Proceedings of the IEEE. pp. 5364-5367.

[4] Sproat, R., 2010. *Lightly Supervised Learning of Text Normaliza-*

tion: Russian Number Names. IEEE Workshop on Spoken Language Technology, Berkeley, CA, 2010.

[5] Jonnalagadda and Topham. 2010. *NEMO: Extraction and normalization of organization names from PubMed affiliations*. J Biomed Discov Collab (2010) vol. 5 pp. 50-75.

[6] Ning Xia, Hongfei Lin, Zhihao Yang, Yanpeng Li, 2011 *Combining multiple disambiguation methods for gene mention normalization*. Expert Systems with Applications, Volume 38, Issue 7, July 2011, Pages 7994-7999

[7] Schlippe, T., Zhu, C., Gebhart, J., Schultz, T., 2010. *"Text Normalization based on Statistical Machine Translation and Internet User Support"*. Interspeech. 2010.

[8] Brody, S., Diakopoulos. N., 2011. *Coooollllll!!!!!!!! Using Word Lengthening to Detect Sentiment in Microblogs*. EMNLP'11.

[9] Han B., and Baldwin, T., 2011. *Lexical normalisation of short text messages: Makn sens a #twitter*. ACL 2011.

[10] Aw, et al. 2006. *A phrase-based statistical model for SMS text normalization*. ACL, 2006.

[11] Pennell D., Liu Y., 2011. *A Character-Level Machine Translation Approach for Normalization of SMS Abbreviations*. IJCNLP.

[12] Nouman Azam, JingTao Yao. 2012. *Comparison of term frequency and document frequency based feature selection metrics in text categorization*. Expert Systems with Applications, Volume 39, Issue 5, April 2012, Pages 4760-4768

[13] Lei Shi, Xinming Ma, Lei Xi, Qiguo Duan, Jingying Zhao. 2011. *Rough set and ensemble learning based semi-supervised algorithm for text classification* Expert Systems with Applications, Volume 38, Issue 5, May 2011, Pages 6300-6306

[14] Och J., Ney. H., 2003. *"A systematic comparison of various alignment models"*. Computational Linguistics, Vol. 29, No. 1 pp. 19-51, 2003.

[15] Koehn P., F.J. Och D. Marcu. 2003. *"Statistical Phrase-based translation"*. Human Language Technology Conference 2003 (HLT-NAACL 2003), Edmonton, Canada, pp. 127-133, May 2003.

[16] Stolcke A. 2002. *"SRILM - An Extensible Language Modelling Toolkit"*. ICSLP. 2002. Denver Colorado, USA.

Noise-Robust Voice Conversion Based on Spectral Mapping on Sparse Space

Ryoichi Takashima, Ryo Aihara, Tetsuya Takiguchi, Yasuo Ariki

Graduate School of System Informatics, Kobe University, Japan
takashima@me.cs.scitec.kobe-u.ac.jp, aihara@me.cs.scitec.kobe-u.ac.jp
takigu@kobe-u.ac.jp, ariki@kobe-u.ac.jp

Abstract

This paper presents a voice conversion (VC) technique for noisy environments based on a sparse representation of speech. In our previous work, we discussed an exemplar-based VC technique for noisy environments. In that report, source exemplars and target exemplars are extracted from the parallel training data, having the same texts uttered by the source and target speakers. The input source signal is represented using the source exemplars and their weights. Then, the converted speech is constructed from the target exemplars and the weights related to the source exemplars. However, this exemplar-based approach needs to hold all training exemplars (frames) and it requires high computation times to obtain the weights of the source exemplars. In this paper, we propose a framework to train the basis matrices of source and target exemplars so that they have a common weight matrix. By using the basis matrices instead of the exemplars, the VC is performed with lower computation times than with the exemplar-based method. The effectiveness of this method was confirmed by comparing its effectiveness, in speaker conversion experiments using noise-added speech data, with the effectiveness of an exemplar-based method and a conventional Gaussian mixture model (GMM)-based method.

Index Terms: voice conversion, sparse representation, non-negative matrix factorization, noise robustness

1. Introduction

Voice conversion (VC) is generally a technique for changing specific information in an input speech while maintaining the other information in the utterance, such as its linguistic information. One of the most popular applications using the VC technique is speaker conversion, where an utterance spoken by a source speaker is morphed so that it sounds as if it had been spoken by a specified target speaker. There have also been studies on various tasks, such as emotion conversion ([1, 2]), speaking assistance ([3, 4]), and so on, which make use of VC techniques.

Many statistical approaches to VC have been studied ([5, 6, 7]). Among these approaches, the GMM-based mapping approach [7] is widely used, and a number of improvements have been proposed. Toda et al. [8] introduced dynamic features and the global variance (GV) of the converted spectra over a time sequence. Helander et al. [9] proposed transforms based on partial least squares (PLS) in order to prevent the over-fitting problem of standard multivariate regression. There have also been approaches that do not require parallel data that make use of GMM adaptation techniques [10] or eigen-voice GMM (EV-GMM) ([11, 12]).

However, the effectiveness of these approaches was confirmed with clean speech data, and the utilization in noisy environments was not considered. The noise in the input signal is not only output with the converted signal, but may also degrade the conversion performance itself due to unexpected mapping of source features. Hence, a VC technique that takes into consideration the effect of noise is of interest.

Recently, approaches based on sparse representations have gained interest in a broad range of signal processing. In the field of speech processing, non-negative matrix factorization (NMF) [13] is a well-known approach for source separation and speech enhancement ([14, 15]). In these approaches, the observed signal is represented by a linear combination of a small number of atoms, such as the exemplar and basis of NMF. In some approaches for source separation, the atoms are grouped for each source, and the mixed signals are expressed with a sparse representation of these atoms. By using only the weights of the atoms related to the target signal, the target signal can be reconstructed. Gemmeke et al. [16] also proposes an exemplar-based method for noise robust speech recognition. In that method, the observed speech is decomposed into the speech atoms, noise atoms, and their weights. Then the weights of the speech atoms are used as phonetic scores instead of the likelihoods of hidden Markov models for speech recognition.

In our previous work [17], we discussed an exemplar-based VC technique for noisy environments. In that report, source exemplars and target exemplars are extracted from the parallel training data, having the same texts uttered by the source and target speakers. Also, the noise exemplars are extracted from the before- and after-utterance sections in an observed signal. For this reason, no training processes related to noise signals are required. The input source signal is expressed with a sparse representation of the source exemplars and noise exemplars. Only the weights related to the source exemplars are picked up, and the target signal is constructed from the target exemplars and the picked-up weights. This method showed better performances than the conventional GMM-based method in speaker conversion experiments using noise-added speech data. However, this exemplar-based approach needs to hold all training exemplars (frames) and it requires high computation times to obtain the weights of the source exemplars.

In this paper, we propose a framework to train the basis matrices of source and target exemplars so that they have a common weight matrix. The basis matrix of the source exemplars is trained using NMF, and then the weight matrix of the source exemplars is obtained. Next, the basis matrix of the target exemplars is trained using NMF, where the weight matrix is fixed to that obtained from the source exemplars. By using the basis matrices instead of the exemplars, the VC is performed with lower computation times than with the exemplar-based method. The effectiveness of this method was confirmed by comparing its effectiveness, in speaker conversion experiments using clean speech data and noise-added speech data, with the effectiveness of an exemplar-based method and the conventional Gaussian mixture model (GMM)-based method.

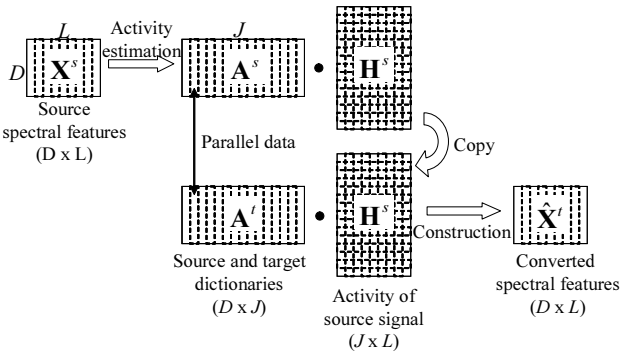


Figure 1: Voice conversion based on the sparse representation

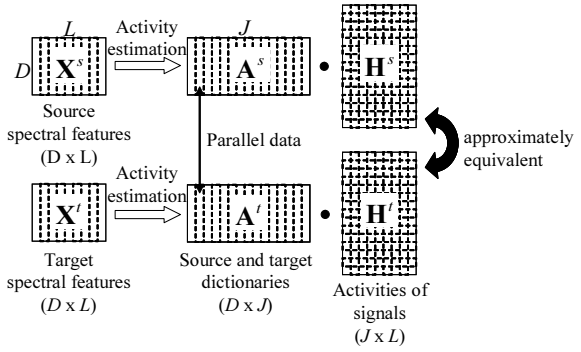


Figure 2: Assumption of the parallelism of source and target dictionaries

2. Voice Conversion Based on Sparse Representation

This section describes a VC method based on the sparse representation [17]. In the approaches based on sparse representations, the observed signal is represented by a linear combination of a small number of atoms.

$$\mathbf{x}_l \approx \sum_{j=1}^{J} \mathbf{a}_j h_{j,l} = \mathbf{A}\mathbf{h}_l \quad (1)$$

$\mathbf{x}_l$ is the l-th frame of the observation. $\mathbf{a}_j$ and $h_{j,l}$ are the j-th atom and the weight, respectively. $\mathbf{A} = [\mathbf{a}_1 \dots \mathbf{a}_J]$ and $\mathbf{h}_l = [h_{1,l} \dots h_{J,l}]^T$ are the collection of the atoms and the stack of weights. When the weight vector $\mathbf{h}_l$ is sparse, the observed signal can be represented by a linear combination of a small number of atoms that have non-zero weights. In this paper, the collection of atoms $\mathbf{A}$ and the weight vector $\mathbf{h}_l$ are called '*dictionary*' and '*activity*', respectively. For the frame sequence data $\mathbf{X} = [\mathbf{x}_1 \dots \mathbf{x}_L]$, Eq. (1) is expressed as the inner product of two matrices.

$$\mathbf{X} \approx \mathbf{A}\mathbf{H} \quad (2)$$

$$\mathbf{X} = [\mathbf{x}_1 \dots \mathbf{x}_L], \quad \mathbf{H} = [\mathbf{h}_1 \dots \mathbf{h}_L] \quad (3)$$

L is the number of the frames.

Figure 1 shows the schema of the VC method based on the sparse representation. D, L, J are the numbers of dimensions, frames and atoms, respectively. In this method, the parallel dictionaries, which consist of source and target dictionaries having the same size, are used to map the source signal to the target one. The parallel dictionaries are structured from the parallel training data, which have the same texts uttered by the source and target speakers, and they are aligned using dynamic programming (DP) matching.

This method assumes that when the source signal and the target signal are expressed with sparse representations of the source dictionary and the target dictionary, respectively, then, the obtained activity matrices are approximately equivalent as shown in Figure 2. Based on this assumption, the activity of the source signal estimated with the source dictionary can be substituted for that of the target signal. Therefore, as shown in Figure 1, the input source signal is represented using the source dictionary and the activity. Then, the converted speech is constructed from the target dictionary and the activity related to the source dictionary.

This VC method can be combined with an NMF-based noise reduction method. Then, the noise dictionary is extracted from the before- and after-utterance sections in an observed signal, and the noise dictionary is concatenated with the source dictionary. The noisy source signal is expressed with a sparse representation of the source dictionary and noise dictionary. Only the weights related to the source dictionary are picked up, and the target signal is constructed from the target dictionary and the picked-up weights.

However, this exemplar-based approach defines the parallel dictionary with the parallel training data themselves. Hence, this method needs to hold all training exemplars (frames) and it requires high computation times to obtain the weights of the source exemplars. In conventional NMF-based noise reduction methods, the dictionary $\mathbf{A}$ is not defined with the training exemplars, but with much fewer bases. These bases are trained using the NMF in advance. However, when the basis matrices of source exemplars and target exemplars are trained using the NMF independently, the parallelism of the source and target dictionaries shown in Figure 2 is lost.

Therefore, in this paper, we propose a framework to train the basis matrices of source and target exemplars so that they have a common weight matrix. By using the basis matrices instead of the exemplars, the VC is performed with lower computation times than with the exemplar-based method.

3. Proposed Method

3.1. Training of the Parallel Basis Matrices

This section describes the framework to train the basis matrices of source and target exemplars. We optimize the source basis matrix $\mathbf{A}^s$ and target basis matrix $\mathbf{A}^t$ so that when the source signal and target signal are expressed with sparse representations of $\mathbf{A}^s$ and $\mathbf{A}^t$, respectively, the obtained activity matrices are equivalent, as shown in Figure 2.

Table 1 shows the algorithm of the training of the parallel basis matrices. At first, for the training source data (exemplars) $\mathbf{X}^s$, the basis matrix $\mathbf{A}^s$ and the activity matrix $\mathbf{H}^s$ are optimized using the NMF with the sparse constraint [16]. In the framework of the NMF with the sparse constraint, it minimizes the following cost function:

$$d(\mathbf{X}^s, \mathbf{A}^s\mathbf{H}^s) + ||(\lambda \mathbf{1}^{(1\times L)}) .* \mathbf{H}^s||_1 \quad s.t. \quad \mathbf{A}^s,\ \mathbf{H}^s \geq 0. \quad (4)$$

Here, $.*$ and $\mathbf{1}$ are an element-wise multiplication and an all-one vector, respectively. The first term is the Kullback-Leibler (KL) divergence between $\mathbf{X}^s$ and $\mathbf{A}^s\mathbf{H}^s$. The second term is the sparse constraint with the L1-norm regularization term that

Table 1: Algorithm of the training of the parallel basis matrices

Training of source basis matrix $\mathbf{A}^s$
• Set source training exemplars to $\mathbf{X}^s$
• Optimize $\mathbf{A}^s$ and $\mathbf{H}^s$ by Eq. (5) and (6)
Training of target basis matrix $\mathbf{A}^t$
• Set target training exemplars to $\mathbf{X}^t$
• Fix the activity matrix to $\mathbf{H}^s$, and optimize $\mathbf{A}^t$ by Eq. (8)

causes $\mathbf{H}^s$ to be sparse. λ is the weight of the sparse constraint. $\mathbf{A}^s$ and $\mathbf{H}^s$ minimizing (4) are estimated iteratively applying the following update rules:

$$\mathbf{A}^s_{n+1} = \mathbf{A}^s_n .* (\mathbf{H}^s_n(\mathbf{X}^s ./ \mathbf{A}^s_n\mathbf{H}^s_n)^T ./ \mathbf{H}^s_n\mathbf{1}^{(1\times D)})^T \quad (5)$$

$$\mathbf{H}^s_{n+1} = \mathbf{H}^s_n .* (\mathbf{A}^{s^T}_n (\mathbf{X}^s ./ (\mathbf{A}^s_n\mathbf{H}^s_n))) ./ (\mathbf{A}^{s^T}_n\mathbf{1}^{(J\times L)} + \lambda\mathbf{1}^{(1\times L)}) \quad (6)$$

where $./$ and $\mathbf{1}$ are an element-wise division and an all-one matrix, respectively.

Next, using the activity matrix $\mathbf{H}^s$ obtained by Eq. (6), the target basis matrix $\mathbf{A}^t$ of the training target exemplars $\mathbf{X}^t$ is optimized. Then, $\mathbf{A}^t$ is optimized so that the activity matrix is equivalent to $\mathbf{H}^s$, i.e. $\mathbf{A}^t$ is optimized to minimize the following cost function:

$$d(\mathbf{X}^t, \mathbf{A}^t\mathbf{H}^s) \quad s.t. \quad \mathbf{A}^t \geq 0. \quad (7)$$

In this optimization, the activity matrix is fixed to $\mathbf{H}^s$, and only $\mathbf{A}^t$ is updated by the following update rule:

$$\mathbf{A}^t_{n+1} = \mathbf{A}^t_n .* (\mathbf{H}^s(\mathbf{X}^t ./ \mathbf{A}^t_n\mathbf{H}^s)^T ./ \mathbf{H}^s\mathbf{1}^{(1\times D)})^T. \quad (8)$$

3.2. Voice Conversion of Noisy Source Signal

3.2.1. Estimation of Activity from Noisy Source Signal

From the before- and after-utterance sections in the observed (noisy) signal, the exemplars (frames) of the noise are extracted, and the noise dictionary is structured from the noise exemplars for each utterance. For this reason, no training processes related to noise signals are required. In the approach based on the sparse representation, the spectrum of the noisy source signal at frame l is approximately expressed by a non-negative linear combination of the source dictionary, noise dictionary, and their activities.

$$\begin{aligned}\mathbf{x}_l &= \mathbf{x}^s_l + \mathbf{x}^n_l \\ &\approx \sum_{j=1}^{J}\mathbf{a}^s_j h^s_{j,l} + \sum_{k=1}^{K}\mathbf{a}^n_k h^n_{k,l} \\ &= [\mathbf{A}^s\mathbf{A}^n]\begin{bmatrix}\mathbf{h}^s_l \\ \mathbf{h}^n_l\end{bmatrix} \quad s.t. \quad \mathbf{h}^s_l, \mathbf{h}^n_l \geq 0 \\ &= \mathbf{A}\mathbf{h}_l \quad s.t. \quad \mathbf{h}_l \geq 0 \end{aligned} \quad (9)$$

$\mathbf{x}^s_l$ and $\mathbf{x}^n_l$ are the magnitude spectra of the source signal and the noise, respectively. $\mathbf{A}^s$, $\mathbf{A}^n$, $\mathbf{h}^s_l$ and $\mathbf{h}^n_l$ are the source dictionary (basis matrix) trained by Eq. (5), noise dictionary (exemplars), and their activities at frame l, respectively. Given the spectrogram, (9) can be written as follows:

$$\begin{aligned}\mathbf{X} &\approx [\mathbf{A}^s\mathbf{A}^n]\begin{bmatrix}\mathbf{H}^s \\ \mathbf{H}^n\end{bmatrix} \quad s.t. \quad \mathbf{H}^s, \mathbf{H}^n \geq 0 \\ &= \mathbf{A}\mathbf{H} \quad s.t. \quad \mathbf{H} \geq 0. \end{aligned} \quad (10)$$

In order to consider only the shape of the spectrum, $\mathbf{X}$, $\mathbf{A}^s$ and $\mathbf{A}^n$ are first normalized for each frame, basis or exemplar so that the sum of the magnitudes over frequency bins equals unity.

$$\begin{aligned}\mathbf{M} &= \mathbf{1}^{(D\times D)}\mathbf{X} \\ \mathbf{X} &\leftarrow \mathbf{X}./\mathbf{M} \\ \mathbf{A} &\leftarrow \mathbf{A}./(\mathbf{1}^{(D\times D)}\mathbf{A}) \end{aligned} \quad (11)$$

The joint matrix $\mathbf{H}$ is estimated based on NMF with the sparse constraint that minimizes the following cost function:

$$d(\mathbf{X}, \mathbf{A}\mathbf{H}) + ||(\lambda\mathbf{1}^{(1\times L)}) .* \mathbf{H}||_1 \quad s.t. \quad \mathbf{H} \geq 0. \quad (12)$$

The weights of the sparsity constraints can be defined for each basis and exemplar by defining $\lambda^T = [\lambda_1 \ldots \lambda_J \ldots \lambda_{J+K}]$. In this paper, the weights for source bases $[\lambda_1 \ldots \lambda_J]$ were set to 0.15, and those for noise exemplars $[\lambda_{J+1} \ldots \lambda_{J+K}]$ were set to 0. $\mathbf{H}$ minimizing (12) is estimated iteratively applying the following update rule:

$$\mathbf{H}_{n+1} = \mathbf{H}_n .* (\mathbf{A}^T(\mathbf{X}./(\mathbf{A}\mathbf{H}))) ./ (\mathbf{1}^{((J+K)\times L)} + \lambda\mathbf{1}^{(1\times L)}). \quad (13)$$

3.2.2. Target Speech Construction

From the estimated joint matrix $\mathbf{H}$, the activity of source signal $\mathbf{H}^s$ is extracted, and by using the activity and the target dictionary, the converted spectral features are constructed. Then, the target dictionary is also normalized for each basis in the same way the source dictionary was.

$$\mathbf{A}^t \leftarrow \mathbf{A}^t./(\mathbf{1}^{(D\times D)}\mathbf{A}^t) \quad (14)$$

$\mathbf{A}^t$ is the target dictionary (basis matrix) trained by Eq. (8). Next, the normalized target spectral feature is constructed, and the magnitudes of the source signal calculated in (11) are applied to the normalized target spectral feature.

$$\hat{\mathbf{X}}^t = (\mathbf{A}^t\mathbf{H}^s) .* \mathbf{M} \quad (15)$$

In this paper, the input source feature is expressed using the magnitude spectrum calculated by STFT because the magnitude spectrum is compatible with the NMF-based noise reduction. On the other hand, the converted spectral feature is expressed as a STRAIGHT spectrum [18] that is compatible with the speech synthesis. The target speech is synthesized using a STRAIGHT synthesizer. Then, F0 information is converted using a conventional linear regression based on the mean and standard deviation.

4. Experiments

4.1. Experimental Conditions

The proposed VC technique was evaluated by comparing it with an exemplar-based method [17] and a conventional GMM-based method [7] in a speaker conversion task using clean speech data and noise-added speech data. The source speaker and target speaker were one male and one female speaker, whose speech is stored in the ATR Japanese speech database, respectively. The sampling rate was 8 kHz.

Two hundred sixteen words of clean speech were used to construct parallel dictionaries in the methods based on the sparse representation and used to train the GMM in GMM-based method. In the exemplar-based method, the number of

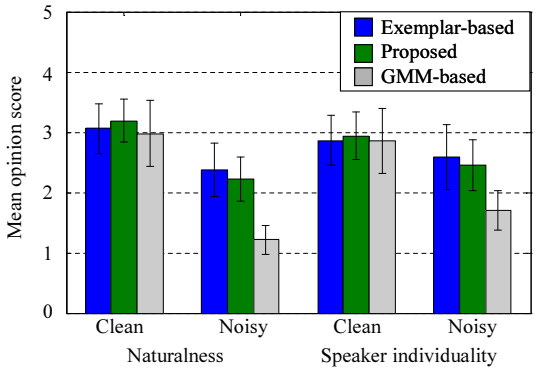


Figure 3: Mean opinion scores (MOS) for each method

exemplars of source and target dictionaries was 58,426. Then, in our proposed method, 1,000 bases were trained from the exemplars for each dictionary. Twenty-five sentences of clean speech or noisy speech were used to evaluate. The noisy speech was created by adding a noise signal recorded in a restaurant (taken from the CENSREC-1-C database) to the clean speech sentences. The SNR was 15 dB. The noise dictionary is extracted from the before- and after-utterance section in the evaluation sentence. The average number of exemplars in the noise dictionary for one sentence was 110.

In the methods based on the sparse representation, a 257-dimensional magnitude spectrum was used as the feature vectors for input signal, source dictionary and noise dictionary, and a 513-dimensional STRAIGHT spectrum was used for the target dictionary. The number of iterations used to estimate the activity was 500. In the GMM-based method, the 1^{st} through 40^{th} linear-cepstral coefficients obtained from the STRAIGHT spectrum were used as the feature vectors. The number of mixtures was 64.

4.2. Experimental Results

We performed an opinion test on the naturalness and speaker individuality of the converted speech. In the opinion test, the opinion score was set to a 5-point scale (5: excellent, 4: good, 3: fair, 2: poor, 1: bad). The tests were carried out with 7 subjects. For the evaluation of naturalness, each subject listened to the converted speech and evaluated how natural the sample sounded. For the evaluation of speaker individuality, each subject listened to the target speech. Then the subject listened to the converted speech and evaluated how similar the converted speech and the target one.

Figure 3 shows the mean opinion scores (MOS) for each method. The error bars show 95% confidence intervals. As shown in this figure, when clean speech data was used, the performances of the three methods were not so different in both evaluation criteria. However, when noisy speech data was used, the performances of GMM-based method degraded considerably especially in naturalness. This might be because the noise caused unexpected mapping in the GMM-based method, and the speech was converted with a lack of naturalness. On the other hand, the degradations of the performances of the VC methods based on the sparse representation were less than those of GMM-based method. The performances of the proposed method were slightly lower than that of the exemplar-based method when noisy speech data was used. However, for obtaining the activity matrix, the computation time of the proposed method (about 30 seconds for 1 sentence on Intel Core i7 2.80 GHz personal computer) was about 30 times faster than that of the exemplar-based method (about 910 seconds).

Table 2: Spectral distortion improvement ratio (SDIR) [dB] for noisy speech

	Exemplar-based	Proposed	GMM-based
SDIR [dB]	3.8	3.7	3.2

Table 2 shows the spectral distortion improvement ratio (SDIR) [dB] for noisy input source signal. The SDIR is defined as follows.

$$\mathrm{SDIR[dB]} = 10\log_{10}\frac{\sum_d |\mathbf{X}^t(d) - \hat{\mathbf{X}}^t(d)|^2}{\sum_d |\mathbf{X}^t(d) - \mathbf{X}^s(d)|^2} \quad (16)$$

Here, $\mathbf{X}^s$, $\mathbf{X}^t$ and $\hat{\mathbf{X}}^t$ are normalized so that the sum of the magnitudes over frequency bins equals unity. As shown in this table, the distortion improvements of the methods based on the sparse representation were higher than GMM-based method. The distortion improvements of the proposed method was slightly lower than that of the exemplar-based method.

5. Conclusions

In this paper, we discussed a noise-robust VC technique based on sparse representation. We proposed a framework to train the basis matrices of source and target exemplars so that they have a common activity matrix. The basis matrix of the source exemplars is trained using the NMF. Then, the basis matrix of the target exemplars is trained using the NMF, where the weight matrix is fixed to that obtained from the source exemplars. By using the basis matrices instead of the exemplars, the VC is performed with lower computation times than with the exemplar-based method. When a noisy input signal is converted to the target signal, the noise exemplars are extracted from the before- and after-utterance sections in an observed signal. The noisy signal is expressed with a sparse representation of the source basis matrix and noise exemplars. The target signal is constructed from the target basis matrix and the activity matrix related to the source basis matrix.

In comparison experiments between the proposed method, an exemplar-based method and a conventional GMM-based method, the proposed method showed better performances than GMM-based method when evaluating noisy speech. The performances of the proposed method were slightly lower than that of the exemplar-based method when noisy speech data was used. But for obtaining the activity matrix, the computation time of the proposed method was about 30 times faster than that of the exemplar-based method.

However, the proposed method still requires higher computation times than that of GMM-based method. While our proposed method took about 30 seconds for 1 sentence to convert speech features, the GMM-based method spent about 1 second to do this. In future work, we will investigate the optimal number of bases and evaluate the performances under other noise conditions. We will also try to introduce dynamic information, such as segment features. In addition, this method has a limitation in that it can be applied to only one-to-one voice conversation because it requires parallel speech data having the same

texts uttered by the source and target speakers. Hence, we will investigate a method that does not use parallel data. Future work will also include efforts to study other noise conditions, such as a low-SNR condition, and apply this method to other VC applications.

6. References

[1] H. Kawanami, Y. Iwami, T. Toda, H. Saruwatari, and K. Shikano, "GMM-based voice conversion applied to emotional speech synthesis," in *Proc. INTERSPEECH*, 2003, pp. 2401–2404.

[2] C. Veaux and X. Robet, "Intonation conversion from neutral to expressive speech," in *Proc. INTERSPEECH*, 2011, pp. 2765–2768.

[3] K. Nakamura, T. Toda, H. Saruwatari, and K. Shikano, "Speaking-aid systems using gmm-based voice conversion for electrolaryngeal speech," *Speech Communication*, vol. 54, no. 1, pp. 134–146, 2012.

[4] H. Doi, K. Nakamura, T. Toda, H. Saruwatari, and K. Shikano, "Esophageal speech enhancement based on statistical voice conversion with Gaussian mixture models," *IEICE Trans. Information and Systems*, vol. E93-D, no. 9, pp. 2472–2482, 2010.

[5] M. Abe, S. Nakamura, K. Shikano, and H. Kuwabara, "Voice conversion through vector quantization," in *Proc. ICASSP*, 1988, pp. 655–658.

[6] H. Valbret, E. Moulines, and J. P. Tubach, "Voice transformation using PSOLA technique," *Speech Communication*, vol. 11, no. 2–3, pp. 175–187, 1992.

[7] Y. Stylianou, O. Cappe, and E. Moulines, "Continuous probabilistic transform for voice conversion," *IEEE Trans. Speech and Audio Processing*, vol. 6, no. 2, pp. 131–142, 1998.

[8] T. Toda, A. Black, and K. Tokuda, "Voice conversion based on maximum likelihood estimation of spectral parameter trajectory," *IEEE Trans. Audio, Speech and Language Processing*, vol. 15, no. 8, pp. 2222–2235, 2007.

[9] E. Helander, T. Virtanen, J. Nurminen, and M. Gabbouj, "Voice conversion using partial least squares regression," *IEEE Trans. Audio, Speech and Language Processing*, vol. 18, no. 5, pp. 912–921, 2010.

[10] C. H. Lee and C. H. Wu, "Map-based adaptation for speech conversion using adaptation data selection and non-parallel training," in *Proc. INTERSPEECH*, 2006, pp. 2254–2257.

[11] T. Toda, Y. Ohtani, and K. Shikano, "Eigenvoice conversion based on gaussian mixture model," in *Proc. INTERSPEECH*, 2006, pp. 2446–2449.

[12] D. Saito, K. Yamamoto, N. Minematsu, and K. Hirose, "One-to-many voice conversion based on tensor representation of speaker space," in *Proc. INTERSPEECH*, 2011, pp. 653–656.

[13] D. D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," in *Proc. Neural Information Processing System*, 2001, pp. 556–562.

[14] T. Virtanen, "Monaural sound source separation by non-negative matrix factorization with temporal continuity and sparseness criteria," *IEEE Trans. Audio, Speech and Language Processing*, vol. 15, no. 3, pp. 1066–1074, 2007.

[15] M. N. Schmidt and R. K. Olsson, "Single-channel speech separation using sparse non-negative matrix factorization," in *Proc. INTERSPEECH*, 2006, pp. 2614–2617.

[16] J. F. Gemmeke, T. Viratnen, and A. Hurmalainen, "Exemplar-based sparse representations for noise robust automatic speech recognition," *IEEE Trans. Audio, Speech and Language Processing*, vol. 19, no. 7, pp. 2067–2080, 2011.

[17] R. Takashima, T. Takiguchi, and Y. Ariki, "Exemplar-based voice conversion in noisy environment," in *Proc. SLT*, 2012, pp. 313–317.

[18] H. Kawahara, I. Masuda-Katsuse, and A. de Cheveigne, "Restructuring speech representations using a pitch-adaptive time-frequency smoothing and an instantaneous-frequency-based F0 extraction: Possible role of a repetitive structure in sounds," *Speech Communication*, vol. 27, pp. 187–207, 1999.

Cross-variety speaker transformation in HSMM-based speech synthesis

Markus Toman, Michael Pucher, Dietmar Schabus

Telecommunications Research Center (FTW), Vienna, Austria

{toman,pucher,schabus}@ftw.at

Abstract

We present and compare different approaches for cross-variety speaker transformation in Hidden Semi-Markov Model (HSMM) based speech synthesis that allow for a transformation of an arbitrary speaker's voice from one variety to another one. The methods developed are applied to three different varieties, namely standard Austrian German, one Middle Bavarian (Upper Austria, Bad Goisern) and one South Bavarian (East Tyrol, Innervillgraten) dialect. For data mapping of HSMM-states we use Kullback-Leibler divergence, transfer probability density functions to the decision tree of the other variety and perform speaker adaptation. We investigate an existing data mapping method and a method that constrains the mappings for common phones and show that both methods can retain speaker similarity and variety similarity. Furthermore we show that in some cases the constrained mapping method gives better results than the standard method.

Index Terms: speech synthesis, dialect, transformation, language variety

1. Introduction

Acoustic language transformation has received much interest in the last years [1, 2, 3]. In general, it is the problem of transforming a speaker's voice into another language retaining speaker identity. Language transformation has applications in speech-to-speech translation and education. In this paper we consider a restricted version of the problem where we want to transform a speaker's acoustic model into a model of the same speaker in a different variety of the same language. A variety can be a dialect, sociolect, or accent. Here we apply variety transformation to standard and dialect.

Although being a simpler problem, it still has an interesting range of possible applications like language learning. If a user wants to learn a certain variety V_{learn} (dialect, sociolect, accent), a variety transformation system can transform his or her variety V_{user} into the target variety V_{learn}. One can then use samples of this voice for learning and comparison with speech that is produced by the user. The main application of such a system would consist in teaching the standard variety to speakers with non-standard varieties but of course it is also possible to use the transformation in the other direction. In this paper we consider transformations between dialects, from the standard to the dialect and from dialect to standard.

We have previously shown how to achieve a one-way transformation between standard and dialect [4]. In this paper, we consider one standard variety and two different dialects and perform all possible transformations. The modeling techniques developed here can further be applied to accented speech. As a first method, we implemented a data mapping approach that is described below and in [4]. We extend this data mapping approach by a constraint-based approach where we map only between models that are in an overlapping phone set. The basic idea of the second method is to exclude mappings where the phones are common in the phone set but the representative phones of the mapped models are different.

2. Data

In previous and current projects we recorded and annotated phonetically balanced speech data in two dialectal Austrian varieties from Innervillgraten in East Tyrol (IVG) and from Bad Goisern in Upper Austria (GOI), as well as standard Austrian German (AT). The data acquisition process is described in [5, 6]. The main problems of recording dialect speech are the missing orthographic standard that define how speech is produced from a written form, the missing linguistic resources, and the speaker selection. In this paper we focus on transformations between varieties.

Table 1: *Non-existent phones in mapping.*

	Target variety		
	AT	IVG	GOI
AT	-	49/90 (0.54)	58/100 (0.58)
IVG	35/78 (0.45)	-	23/100 (0.23)
GOI	36/78 (0.46)	23/90 (0.26)	-

Table 1 describes the phone set relations. If we want to tranform AT to IVG, for example, we have to model 49 of the 90 phones in IVG that are nonexistent in AT. The higher the ratio in the table, the more difficult we expect the corresponding variety transformation pair. We can also see that the ratio of missing phones is larger when we have the standard (AT) as the source variety because the phone overlap between the two dialects (GOI, IVG) is larger than the overlap between the standard and either of the two dialects.

3. Speaker-adaptive acoustic modeling

For speaker-adaptive acoustic modeling we used a version of the HSMM-based speech synthesis system (HTS) as published by the EMIME project [7] for our experiments. The input to the system in the training phase is a training set of speech signal waveforms and corresponding full-context label files. These labels contain symbolic representations (phones) of the speech signal as well as contextual information like phonetic and linguistic features. Using this input, speaker-adaptive Hidden semi-Markov average voice Models (HSMMs) are trained for all varieties. In the synthesis phase, labels from a test set are used to generate a synthesized speech signal from the trained models. Methods from text analysis can be used to generate new labels. Multiple speakers can be combined in an average

voice model. Speaker adaptation can then be used to derive a speaker-specific model from this average voice model [8].

Five-state HSMMs are employed in our experiments. We extract 40 mel-frequency cepstral coefficients, fundamental frequency F_0 (modeled as multi-space probability distribution [9]), and a set of 25 band-limited aperiodicity measures from the speech signal. Dynamic features were used to improve continuity of the generated speech spectra [10]. The decision-tree based context clustering technique as described in [11] and as available in HTS has been used to share model parameters across multiple contexts. We use different sets of decision tree questions for each variety. These are partially handcrafted as well as automatically generated from our phone set definitions.

4. Speaker-adaptive cross-variety transformation

Here we present a cross-variety transformation system that is based on a speaker-adaptive HSMM-based speech synthesis system [7]. The system uses average voice models of the source and target varieties for cross-variety transformation.

Based on the state-level transformation described in [2], we integrated a state mapping mechanism into our cross-variety adaptation system. Using data from multiple speakers in varieties V_1, for which also adaptation data exist, and V_2, to which the voice model should be transformed, we train average voice models [8], denoted as AVG_1 and AVG_2, respectively. The decision trees for those models will then be denoted as DT_1 and DT_2, respectively. DT_1 and DT_2 actually consist of multiple trees for mel-frequency cepstral coefficients, $F0$, aperiodicity and duration for each of the five HSMM states. Since AVG_1 and AVG_2 were trained on different data with different (albeit overlapping) phone sets, they also have a different decision tree structure.

4.1. Data mapping

For every probability density function (pdf) $A \in AVG_1$, we find a pdf $B \in AVG_2$ which minimizes the Kullback-Leibler divergence (KLD),

$$M(A) = \underset{B \in AVG_2}{\arg\min}\ \mathrm{KLD}(A, B), \tag{1}$$

which defines a mapping function M from AVG_1 to AVG_2.

In Figure 1, an illustration of the relation between decision tree, pdf and KLD-mapping can be seen. For example, "mcep_s2_12" refers to the 40-dimensional pdf number 12 for the mel-frequency cepstral coefficients in HSMM state 2. The decision tree questions used in this illustration consist of two parts. The second part is a phonetic symbol from our phone set definitions, for example "t" as in "hat". The first part of the question can be "C" for center, "L" for left and "R" for right, referring to the position of the phone in question.

In the actual system, multiple trees for the feature streams for each HSMM state have been used, resulting in 15 decision trees and 5 additional decision trees for duration modeling. Also, a typical set of questions for the decision tree in our case consists of 1,700 different questions. For example, we calculated the mapping between an Austrian German (AT) average voice and an Innervillgraten (IVG) average voice. This mapping consisted of 13,808 pdf pairs. Therefore, the Austrian German decision trees have 13,808 leaf nodes, making vivid visualizations and manual analysis difficult.

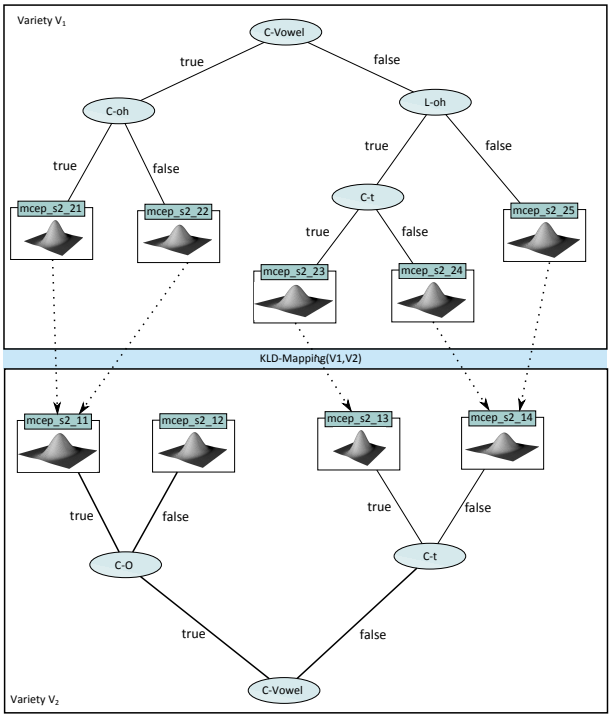


Figure 1: *KLD-Mapping between probability density functions clustered by decision tree.*

Using mapping function M, we map the pdfs of the speaker to be adapted from V_1 to V_2. This is implemented as described in [2, 4].

4.2. Constrained mapping

In addition to the data mapping approach described above, we also investigated a constrained mapping approach. This is based on the idea that mappings should only be made between the same phones, if existent in both varieties. We can however not fulfill this constraint directly, since the mapping is not defined on the phoneme level but only on the model (pdf) level.

To implement the constraint, we apply the following algorithm:
For each pdf $A \in AVG_1$ of a certain variety V_1, we define the representative phone $r(A)$ as the center phone that is most common in the associated full-context labels. Furthermore, we define $P(A)$ as the list of all center phones in all labels used to train A. Figure 2 illustrates how $r(A)$ and $P(A)$ are defined. First we find all full-context labels that have been used to train the corresponding pdf. $P(A)$ is then the set of all center phones from this list of labels and $r(A)$ is the center phone occurring most often. Another possible method to calculate $r(A)$ would be to weigh each label with the number of associated samples, as these have greater influence on the pdf estimation. Next we constrain the mappings on the set of common phones. If the representative phone $r(A)$ occurs not only in phone set P_1 of variety V_1 (as it does per definition) but also in phone set P_2 of variety V_2, then we only map from $A \in AVG_1$ to $B \in AVG_2$, if $r(A)$ is in $P(B)$.

So the common phone data mapping $M(A) = B$ from

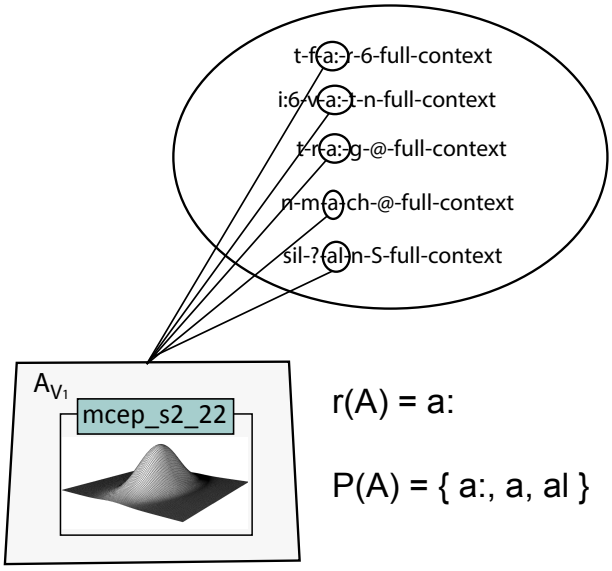


Figure 2: *Definition of representative phone $p(A_{V_1})$ and list of all center phones $pa(A_{V_1})$.*

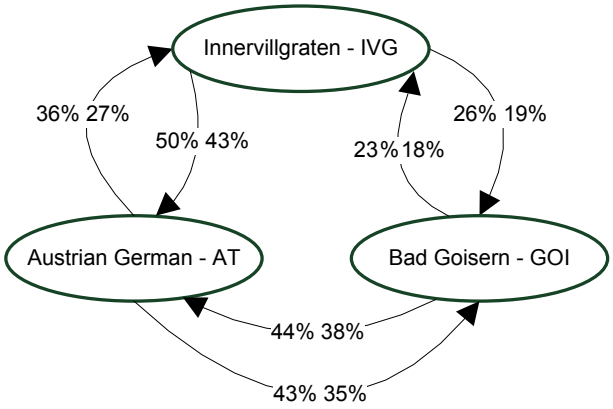


Figure 3: *Transformations between varieties.*

Table 2: Results of the variety similarity judgment part of the evaluation.

Compared methods	wins	ties
DM : CPDM	31 : 27	82

AVG_1 to AVG_2 must fulfill the conditional constraint given in Equation 2:

$$(r(A) \in P_2) \Rightarrow (r(A) \in P(B)) \quad (2)$$

In other words, if the representative phone $r(A)$ occurs in both varieties, we discard all potential mappings $M(A) = B$ for which $r(A)$ is not in the training data of B. If $r(A)$ does not occur in both varieties, we keep all potential mappings (A, B). Of the remaining potential mappings, the mapping with the lowest KLD value as in Equation 1 is selected for further processing.

A possibility to make the constraint stronger would be to require that $r(A)$ is also the most common phone in P_2, so $r(A) = r(B)$. However, we did not evaluate this stronger constrained variant.

Figure 3 shows the percentage of 1-best and 200-best mappings between the different varieties that fulfill the constraint given by Equation 2. n-best means the n mappings with the lowest KLD score. To map from AT to IVG, for example, there are 36% in the 1-best lists and 27% in the 200-best list where the mapping is between equal phones if there is an overlap. The relations here are similar to the nonexistent phones given in Table 1. The smaller the phone overlap, the fewer mappings fulfill the constraint.

4.3. Regression tree generation

For generating the regression tree, we use the algorithm as described previously [4]. To build the regression tree, we delete leaf nodes from DT_2 and move their associated labels to their parent node until the number of adaptation labels associated to every leaf node is above a certain threshold. As these leaf nodes then form the regression classes, this method assures that every regression class contains a certain amount of adaptation data for the calculation of the transformation.

We place labels from the data set of V_1 into the leaf nodes of DT_2 not according to their decision tree questions but to their associated mapped pdfs. Using Figure 1 as an example, a label from V_1 that would be placed in "mcep_s2_22" in DT_1 will be part of the node "mcep_s2_11" in decision tree DT_2. Again note that each label has one pdf associated for each available decision tree, so this process is repeated on multiple trees.

We modified the regression tree building method of HTS to reflect this strategy.

5. Evaluation

We conducted a subjective and an objective evaluation as well as an analysis of specific cases. These will be described in the following sections.

5.1. Subjective evaluation

To evaluate the two methods *Data Mapping* (DM) and *Common Phone Data Mapping* (CPDM), we have carried out a subjective listening evaluation with 27 listeners participating (17 males and 10 females, aged 20 to 55, mean age 28.95), all native German speakers from different regions in Austria, including 9 listeners from our target regions (East Tyrol or Upper Austria). The evaluation consisted of two parts. In the first part we compared synthesized samples from the two methods with a reference signal, and asked the listeners which synthesized sample they found to be more similar to the reference signal in terms of variety. The reference signal was a recording of the same sentence spoken by a (different) speaker of the target variety. We assume that this experiment design allows that listeners who are not themselves speakers of the target variety can still judge the variety similarity. The results in Table 2 show that method DM was considered more similar 31 times, CPDM was considered more similar 27 times and 82 times they were considered equally similar to the reference. The difference in the number of "wins" (31 vs. 27) is not statistically significant according to a Bonferroni-corrected Pearson's χ^2-test of independence ($p > 0.58$). This and the large number of "ties" suggest that none of the two methods is superior to the other.

Additionally, we asked the listeners to specify the degree of similarity concerning variety for the "winning" method (or both methods, in case of a tie), by choosing one of the five options

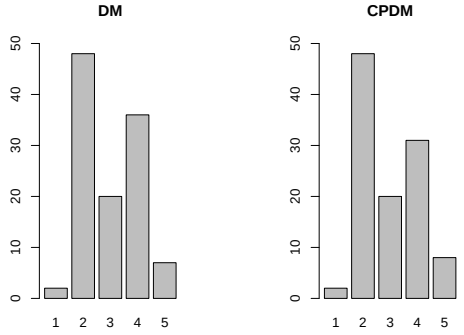


Figure 4: *Frequencies of variety similarity votes for the two methods. 1 means "very similar" and 5 means "very different".*

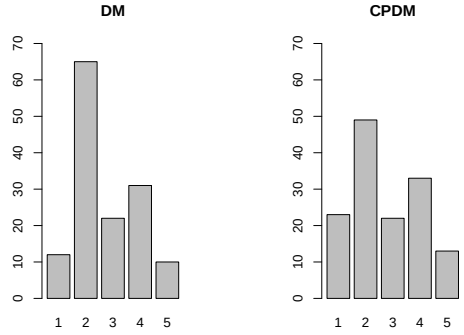


Figure 5: *Frequencies of speaker similarity votes for the two methods. 1 means "very similar" and 5 means "very different".*

Table 3: *Results of the speaker identification part of the evaluation.*

Method	correct	wrong	undecided	sig.
DM	91	35	14	*
CPDM	91	28	21	*

"very similar", "similar", "no opinion", "different" and "very different". The results are given in Figure 4 as frequency bar plots, where 1 means "very similar" and 5 means "very different". We can see that "similar" was the most frequently chosen option. The number of votes for "different" can be explained by the difficulty of the concept variety similarity, which often includes a factor of authenticity. Authenticity can be affected negatively by the overall quality of synthetic speech. Furthermore, there are listeners who are not native speakers of the target varieties.

In the second part of the evaluation, the goal was to assess speaker similarity. We showed the listeners one synthesized sample from one of the two methods and two recorded reference samples. The two references were both the same utterance in a variety different from the target variety of the synthesized sample, one from the target speaker and one from a randomly selected different speaker. The listeners were asked to decide to which of the two references the synthesized sample sounded more similar in terms of speaker identity. The results are given in Table 3, where for each of the two methods, the number of correct, wrong and undecided judgments are presented. For both methods, the number of correct speaker identifications is statistically significantly higher than the number of wrong speaker identifications (Bonferroni-corrected Pearson's χ^2-test of independence with $p < 0.001$).

Again, the listeners were also asked to specify the degree of similarity by choosing one of the five options "very similar", "similar", "no opinion", "different" and "very different". The results are given in Figure 5 as frequency bar plots, where 1 means "very similar" and 5 means "very different". It can be seen that while the number of votes for "similar" decreased for CPDM compared to DM, the number of votes for "very similar" increased.

Table 4: *Results of the objective evaluation.*

	DM		CPDM		SD	
Speaker	μ	σ	μ	σ	μ	σ
IVG 1	6.33	0.43	6.29	0.36	5.42	0.31
IVG 2	7.01	0.72	6.68	0.66	5.46	0.29
GOI 1	6.85	0.43	6.86	0.44	5.90	0.47
GOI 2	7.03	0.74	7.12	0.73	6.10	0.81

5.2. Objective evaluation

We also conducted an objective evaluation by calculating mel-cepstral distortion between the trajectories resulting from the presented methods and trajectories extracted from original recordings. This was possible as we had recordings of standard as well as dialect from some of the speakers. For the analysis we used our AT test set consisting of 23 utterances. We transformed two IVG and two GOI speakers to AT and calculated mel-cepstral distortion between the synthesized results and the AT recordings of the same speaker for the same utterance. The samples were synthesized using the phone durations obtained by automatic alignment of the test recordings.

Table 4 shows the result of this analysis for the four speakers. It can be seen that the mean mel-cepstral distortion is lower for CPDM than for DM for the IVG speakers while it is higher for the GOI speakers. However, only the difference for speaker IVG 2 is significant according to a Bonferroni-corrected Paired t-test ($p < 4 \times 10^{-8}$). This shows that CPDM improves the model for one speaker and does not corrupt the model for the others.

We also trained speaker-dependent (SD) models (using 223 utterances) in AT for every speaker for reference. The mean values and standard deviations for the speaker dependent models compared to the recordings can also be seen in Table 4. As expected, all speaker-dependent AT models have significantly ($p < 2 \times 10^{-14}$) lower mel-cepstral distortion compared to the cross-variety transformation models. This shows that the mel-cepstral distortion metric covers aspects of speaker similarity.

5.3. Analysis of specific cases

When manually inspecting the synthesized waveforms, we noticed structures that remarkably differed for the DM and

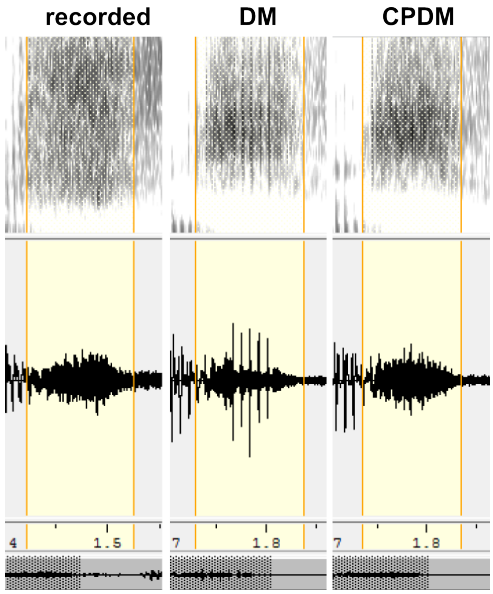


Figure 6: *Waveforms for "s" as recorded and synthesized using DM and CPDM.*

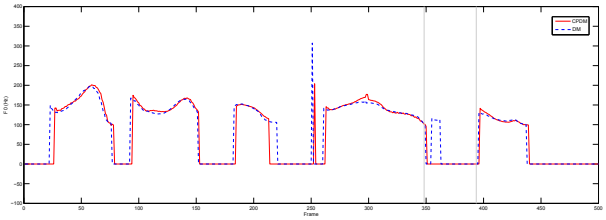


Figure 7: *Different F0 trajectory for DM and CPDM.*

CPDM methods. When listening to these parts, we could find glitches in DM that were absent in CPDM. While these glitches were quite distinct, they did not seem to be the predominant factor to influence scoring in the subjective listening test.

For an example, consider Figure 6. The highlighted section of the waveform corresponds to the main part of the phone "s" and is presented for the recording of a GOI speaker and an AT speaker transformed to GOI using DM and CPDM method.

It can be seen that the waveform generated by CPDM is smoother and more closely resembles the waveform of the natural "s". In the DM version, the "s" has a crackling effect that is absent in the CPDM synthesis. Analyzing the different trajectories for this sample resulted in Figure 7. The "s"-sound is highlighted and it can be seen that the f0 values differ in this region. DM produces a voiced region compared to the correct, unvoiced region produced by CPDM. Reasons for this behavior remain subject of further investigation.

Listening samples for some specific cases can be found on the demopage[1].

6. Conclusion and future work

We compared different approaches for cross-variety speaker transformation in HSMM-based speech synthesis. The developed methods were applied to three different varieties. We investigated a standard data mapping method and a mapping method that constrains the mappings for common phones. In the subjective evaluation, we saw that both data mapping methods can retain speaker similarity to a high degree and variety similarity to a smaller degree. In the pairwise comparison we did not see significant differences between the two methods. This conforms with prior work where different data mapping approaches also led only to subtle changes in the results [3].

We performed an objective evaluation for the bi-lingual data of speakers in two varieties. One of four speakers showed a significant improvement in mel-spectral distortion for CPDM over DM.

We also analyzed specific cases and reported one of them in this article. Further analyses would be necessary to gain a deeper understanding of these effects.

[1] Synthesis samples on http://userver.ftw.at/~mtoman/ssw2013/t

7. Acknowledgements

This research was funded by the Austrian Science Fund (FWF): P23821-N23. The Competence Center FTW Forschungszentrum Telekommunikation Wien GmbH is funded within the program COMET Competence Centers for Excellent Technologies by BMVIT, BMWA, and the City of Vienna. The COMET program is managed by the FFG.

8. References

[1] Y. Qian, J. Xu, and F. K. Soong, "A frame mapping based HMM approach to cross-lingual voice transformation", in Proc. ICASSP, Prague, Czech Republic, 2011, pp. 5120-5123.

[2] Y.-J. Wu, Y. Nankaku, and K. Tokuda, "State mapping based method for cross-lingual speaker adaptation in HMM-based speech synthesis", in Proc. INTERSPEECH, Brighton, United Kingdom, 2009, pp. 528-531.

[3] H. Liang and J. Dines. "Phonological Knowledge Guided HMM State Mapping for Cross-Lingual Speaker Adaptation", in Proc. INTERSPEECH, Florence, Italy, 2011, pp. 1825–1828.

[4] M. Toman, M. Pucher, "Structural KLD for cross-variety speaker adaptation in HMM-based speech synthesis", in Proc. SPPRA, Innsbruck, Austria, 2013.

[5] M. Pucher, D. Schabus, J. Yamagishi, F. Neubarth, and V. Strom, "Modeling and interpolation of austrian german and viennese dialect in HMM-based speech synthesis", Speech Communication, 52(2):164–179, 2010.

[6] M. Pucher, N. Kerschhofer-Puhalo, D. Schabus, S. Moosmüller, G. Hofer, "Language resources for the adaptive speech synthesis of dialects", in Proc. SIDG, Vienna, Austria, 2012.

[7] J. Yamagishi and O. Watts, "The CSTR/EMIME HTS system for Blizzard challenge 2010", in Blizzard Challenge Workshop, Kansai Science City, Japan, 2010.

[8] J. Yamagishi, T. Kobayashi, Y. Nakano, K. Ogata, and J. Isogai, "Analysis of Speaker Adaptation Algorithms for HMM-Based Speech Synthesis and a Constrained SMAPLR Adaptation Algorithm", IEEE Transactions on Audio, Speech, and Language Processing 17.1 (Jan. 2009): 66-83, 2009.

[9] K. Tokuda, T. Masuko, N. Miyazaki, and T. Kobayashi, "Hidden Markov models based on multi-space probability distribution for pitch pattern modeling", in Proc. ICASSP, Phoenix, AR, USA, 1999, pp. 229–232.

[10] T. Masuko, K. Tokuda, T. Kobayashi, and S. Imai, "Speech synthesis using HMMs with dynamic features". in Proc. ICASSP, Atlanta, GA, USA, 1996, pp. 389-392.

[11] T. Yoshimura, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, "Simultaneous modeling of spectrum, pitch and duration in HMM-based speech synthesis", in Proc. EUROSPEECH, Budapest, Hungary, 1999, pp. 2347–2350.

Multi-variety adaptive acoustic modeling in HSMM-based speech synthesis

Markus Toman, Michael Pucher, Dietmar Schabus

Telecommunications Research Center (FTW), Vienna, Austria

{toman,pucher,schabus}@ftw.at

Abstract

In this paper we apply adaptive modeling methods in Hidden Semi-Markov Model (HSMM) based speech synthesis to the modeling of three different varieties, namely standard Austrian German, one Middle Bavarian (Upper Austria, Bad Goisern), and one South Bavarian (East Tyrol, Innervillgraten) dialect. We investigate different adaptation methods like dialect-adaptive training and dialect clustering that can exploit the common phone sets of dialects and standard, as well as speaker-dependent modeling. We show that most adaptive and speaker-dependent methods achieve a good score on overall (speaker and variety) similarity. Concerning overall quality there is no significant difference between adaptive methods and speaker-dependent methods in general for the present data set.

Index Terms: speech synthesis, dialect, voice modeling, adaptation

1. Introduction

Speech synthesis is an important part of human-machine communication systems. At present, speech synthesis systems are mostly restricted to standard varieties, which implies a strong limitation on possible applications.

The dialect or accent of a speaker is an important part of the persona of a voice-based user interface since "there is no such thing as a voice user interface with no personality" [1]. Perception of sociolect and dialect influence our evaluation of speakers' attributes like competence, intelligence, and friendliness. Persona is defined as the "standardized mental image of a personality or character that users infer from the applications voice and language choice" [1], where speech synthesis is an essential part of a spoken dialog system's persona.

To build speech synthesis systems that are able to use a range of different varieties it is important that we have methods that allow for a quick development of these voices. Methods based on adaptation are therefore a natural choice. Furthermore, we can exploit the fact that varieties (dialects and sociolects) and standards have an overlapping phone set. This overlap is illustrated in Figure 1 for the varieties of German we consider in this paper. It can be seen that 38 phones are shared across all three varieties. The small phone set overlap reflects the fact that there is a number of dialect phones that are characteristic and mark differences between standard and variety.

The modeling of accented speech data has received some interest in the last years [2, 3, 4] but the modeling of dialects that differ significantly from the standard language in terms of phonetics and the lexicon is still not widely investigated. This is of course also due to the lack of resources and the difficulty to acquire them.

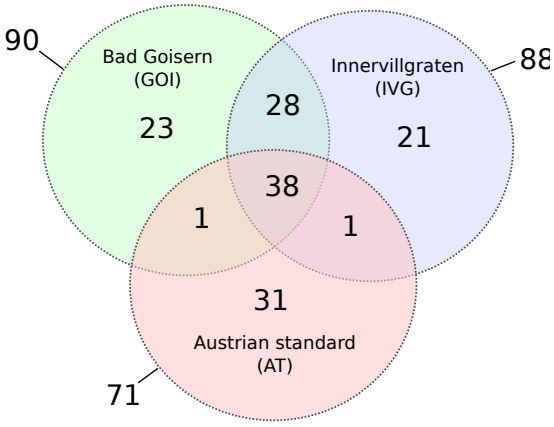


Figure 1: *Overlapping phone sets.*

In [5] it was shown how adaptive dialect modeling methods can be applied to the modeling of two different varieties, namely standard Austrian German and Viennese dialect. Here we show advanced adaptive modeling methods for varieties and evaluate these methods with three Austrian German varieties, namely standard Austrian German and two dialects.

In [6] we investigated cross-variety speaker adaptation between standard Austrian German and the dialects of Innervillgraten and Bad Goisern. This method is based on work of [7] and findings of the EMIME project [8].

2. Speech data and recording

We have recorded and annotated phonetically balanced speech data in different Austrian varieties from Innervillgraten (IVG), Bad Goisern (GOI) [9] and standard Austrian German (AT). In this paper we focus on the modeling of these varieties. The dialects in Austria can be divided into Middle-Bavarian, South-Bavarian, and Alemannic dialects. To cover different regions with as many speakers as possible, we decided to model one Middle-Bavarian and one South-Bavarian dialect in addition to the standard. To restrict the possible variance in the data, we restricted the recordings to a small village in each region: Bad Goisern in Upper Austria for the Middle-Bavarian dialect family and Innervillgraten in East Tyrol for the South-Bavarian dialect family. Initial linguistic studies exist for both dialects [9, 5] but no phone set, corpus, recording script, or synthesizer was available for them.

After a careful phonetic analysis we compiled sets of phonetically balanced sentences (656 for IVG and 665 for GOI) with respect to the phone set established for the dialect, the fre-

Table 1: *Dialect modeling approaches.*

			Data Dependency	
Name	Target	# utt.	Speaker	Dialect
SD-DD (AT)	AT	198	✓	✓
SD-DD (IVG)	IVG	618	✓	✓
SD-DD (GOI)	GOI	622	✓	✓
SI-DD (AT)	AT	1790	×	✓
SI-DD (IVG)	IVG	1236	×	✓
SI-DD (GOI)	GOI	1244	×	✓
SI-SN	AT/IVG/GOI	4270	×	×
SI-SDN	AT/IVG/GOI	4270	×	×
SI-SDNC	AT/IVG/GOI	4270	×	×
DHN	AT/IVG/GOI	4270	×	×

quency of occurrence of each phone in the data, and the context-specific variation of phones. The utterances of the recording script were extracted from a larger corpus of material consisting of 18-20 hours of recordings for each dialect with at least 10 speakers per dialect. These sentences consisted of spontaneous speech (elicited with key words) and translation tasks. We created a lexicon of words occurring in the script. The script was divided into a training and testing part. In the final recordings we recorded 4 speakers (2 male, 2 female) for each dialect. Here we only train models with the male speakers. For our training we have 4 dialect speakers (2 IVG and 2 GOI speakers) where we have dialect and standard data for each speaker, and 1 standard speaker.

The speakers had to fulfill the following linguistic criteria

- "Native speaker", i.e., raised within the dialect
- Consistent application of characteristic phonological processes (e.g., assimilations, deletions)
- Lexical knowledge and morpho-syntactic competence

For recording the dialect data we used a setting where the speaker can hear the utterance he/she is supposed to say and at the same time see an orthographic transcription of the utterance. This is not necessary when an orthographic standard is available and the speakers know how to produce speech from the standard transcription. With this approach we aim to minimize the linguistic variation between the orthographic transcription and the actual spoken utterances of a speaker. Nevertheless, there is still some variation due to fact that speakers may forget what they heard or attempt to correct the reference utterance in case of disagreement.

3. Modeling approaches

Our adaptive speech synthesis system [12] is based on the HSMM-based speech synthesis system (HTS) published by the EMIME project [8]. We use different sets of decision tree questions for each variety. These are partially handcrafted as well as automatically generated from our phone set definitions.

Table 1 defines the modeling approaches that we investigated. SD and SI refer to Speaker-Dependent and Speaker-Independent modeling, DD and DI refer to Dialect-Dependent and Dialect-Independent modeling and SN, SDN, SDNC and DHN refer to Speaker-Normalization, Speaker-Dialect-Normalization, Speaker-Dialect-Normalization with dialect Clustering and Dialect-Hierarchical-Normalization training, respectively. For dialect-dependent modeling, we train average models for each dialect. For dialect-independent modeling, we consider the following approaches: In SI-SN, we train a single model using data from all speakers. SI-SDN means to divide a set of speech data in two varieties uttered by a single speaker (able to speak both varieties) into two subsets of speech data uttered by two different pseudo-speakers (Figure 2). In this example, for speaker 1 AT and IVG recordings exist. Speaker 1 will then be treated as two different speakers, one AT and one IVG speaker. The idea of SI-SDNC is to add dialect information as a context for sub-word units and perform decision-tree-based clustering of dialects in the training of the HSMMs.

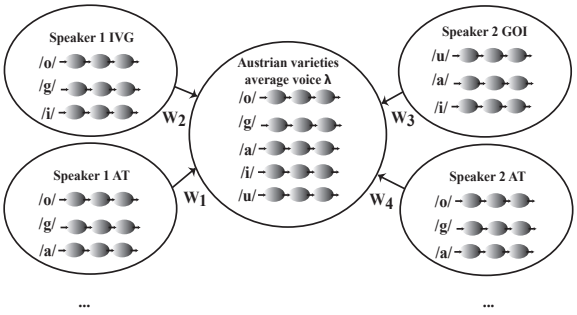


Figure 2: *Speaker-dialect-normalization - SI-SDN.*

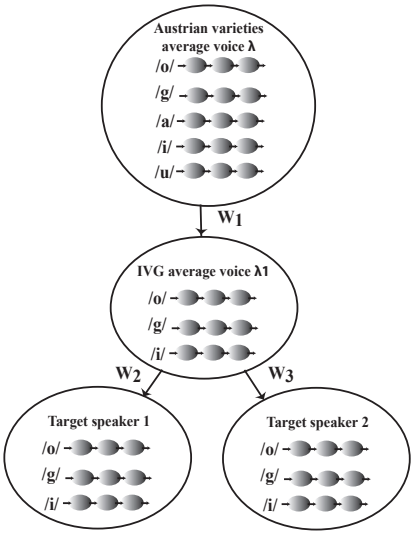


Figure 3: *Dialect-hierarchical normalization - DHN.*

In the clustering of dialects, new questions that identify the variety of an utterance (Is_ivg, Is_goi, Is_at) are added to a set of questions for the decision-tree-based clustering and minimum description length (MDL) based automatic node-splitting [13] is performed. Variety is treated as a clustering context together with other phonetic and linguistic contexts and it is included in the single resulting acoustic model. Note that a decision tree was constructed independently for each combination of state index and acoustic parameter (mel-cepstrum, $\log F_0$, band aperiodicity) and duration. The same idea has been reported for multi-accented English average voice models [14].

In the resulting clusterings, we observe that the first ques-

Table 2: *Occurrences of variety questions in decision trees.*

	# of occurrences				
Feature	State 1	State 2	State 3	State 4	State 5
mel-cepstral	76	139	53	54	67
log F0	28	62	70	44	33
bndap	23	24	37	28	29
duration			70		

tion concerning the variety is used near the roots of the decision trees. Figure 4 shows this part of the constructed decision tree for the mel-cepstral parameters of the third (middle) state and Figure 5 the corresponding duration parameter clustering tree. These are the top-most occurrences of variety questions in the trees and they appear on level 4 in the mel-cepstral-state-3 tree and on level 5 in the duration tree.

Overall occurrences of variety questions in mel-cepstral, logF0 and duration decision trees can be seen in Table 2. It can be seen that variety class questions are relevant in all states. In this example, "Is_ivg" means "Is the current utterance in Innervillgraten dialect?" and "Is_goi" means "Is the current utterance in Bad Goisern dialect?". This means that after these questions, separate Gaussian pdfs are produced for the different dialects. We also observed the labels which have been used to train each single pdf. In SDN only 928 pdfs where estimated using data from a single variety and 1620 pdfs using data from more than one variety. For SDNC, 2431 pdfs were estimated using single variety data and 322 pdfs using data from multiple varieties. This also shows the effect of the the variety questions on the clustering.

In addition to SD-DD, SI-DD, SI-SN, SI-SDN and SI-SDNC, which were already applied for AT and Viennese Dialect (VD) data in the past [5], we also apply dialect-hierarchical normalization (DHN) in this paper. In DHN, a general dialect-independent voice model is trained first, from which then specific dialect-dependent voice models are adapted. Finally, speaker-specific voice models are adapted from these, as shown in Figure 3. Furthermore, we extend this previous work to three different varieties.

We applied model adaptation with AT, IVG, and GOI data to all models. Therefore we have 30 voices in total, where 25 are adapted voices and 5 are speaker- and dialect-dependent voices [1].

4. Evaluation

To assess the quality of the synthetic voices resulting from the different modeling approaches described in Section 3, we have carried out a subjective evaluation with 21 test listeners (8 female, 13 male, aged 20 to 55, mean age 28.95). For each of the three varieties, we have held out 10 test utterances from the training data, in order to allow comparison also to recorded samples, and synthesized each of them using all of the methods for each of our five speakers. Comparing any two models for each (speaker, utterance)-combination gives rise to 1050 comparisons in total, which we distributed among our 21 listeners such that each listener heard each (speaker, utterance)-combination once and each method-pair two to three times.

[1] Synthesis samples on http://userver.ftw.at/~mtoman/ssw2013/m

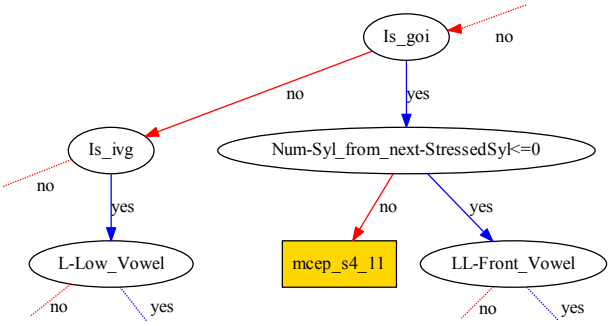


Figure 4: *Dialect clustering results for state 3 of mel-cepstral decision tree.*

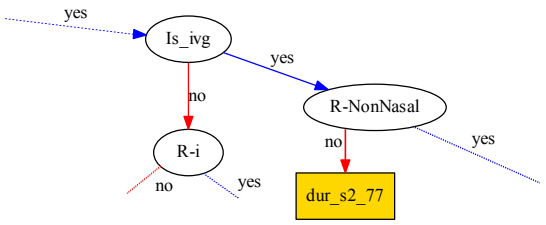


Figure 5: *Dialect clustering results for duration decision tree.*

For each of their 50 comparisons, the listeners heard a recorded reference sample and two samples from two different methods, where all three samples contained the same utterance from the same speaker. After listening to each of the three sound files as many times as they liked, they were asked to tell which of the two samples they felt to be more similar to the reference sample. *Recorded* was also added as a method, i.e., in some comparisons the reference sample and one of the samples in question actually contained the same (recorded) signal. There was also a "tied" option (both samples equally similar to the reference).

The results are given in Table 3, where we have counted the number of "won" comparisons and the number of "ties" for each method pair. In the last column, the symbol "$*$" indicates statistical significance of the preference scores according to Bonferroni-corrected Pearson's χ^2-tests of independence with $p < 0.001$. Even with a relaxed significance threshold of $p < 0.05$, only one additional significance appears (indicated by "$(*)$" in Table 3), but due to the large number of ties in this case (30), we do not consider this a meaningful difference between the two methods SI-SDN and SI-SN.

Additionally, we asked the listeners to specify the degree of similarity for the "winning" method (or both methods, in case of a tie), by choosing one of the five options "very similar", "similar", "no opinion", "different" and "very different". The results are given in Figure 6 as frequency bar plots, where 1 means "very similar" and 5 means "very different".

5. Analysis

We have investigated different adaptive modeling approaches for multi-variety modeling. For the pair-wise comparison of methods we see no significant differences between adapted and speaker-dependent methods, with the exception of dialect-hierarchical training (DHN), which is worse than all other meth-

Table 3: *Subjective pair-wise comparison scores.*

Compared methods			wins	ties	sig.
DHN	:	recorded	1 : 49	0	∗
DHN	:	SD-DD	6 : 26	18	∗
DHN	:	SI-SDN	7 : 26	17	∗
DHN	:	SI-SDNC	5 : 34	11	∗
DHN	:	SI-DD	5 : 34	11	∗
DHN	:	SI-SN	7 : 27	16	∗
recorded	:	SD-DD	50 : 0	0	∗
recorded	:	SI-SDN	49 : 0	1	∗
recorded	:	SI-SDNC	48 : 0	2	∗
recorded	:	SI-DD	46 : 3	1	∗
recorded	:	SI-SN	49 : 0	1	∗
SD-DD	:	SI-SDN	10 : 15	25	
SD-DD	:	SI-SDNC	10 : 15	25	
SD-DD	:	SI-DD	10 : 19	21	
SD-DD	:	SI-SN	19 : 13	18	
SI-SDN	:	SI-SDNC	14 : 8	28	
SI-SDN	:	SI-DD	12 : 15	23	
SI-SDN	:	SI-SN	16 : 4	30	(∗)
SI-SDNC	:	SI-DD	13 : 15	22	
SI-SDNC	:	SI-SN	18 : 15	17	
SI-DD	:	SI-SN	10 : 14	26	

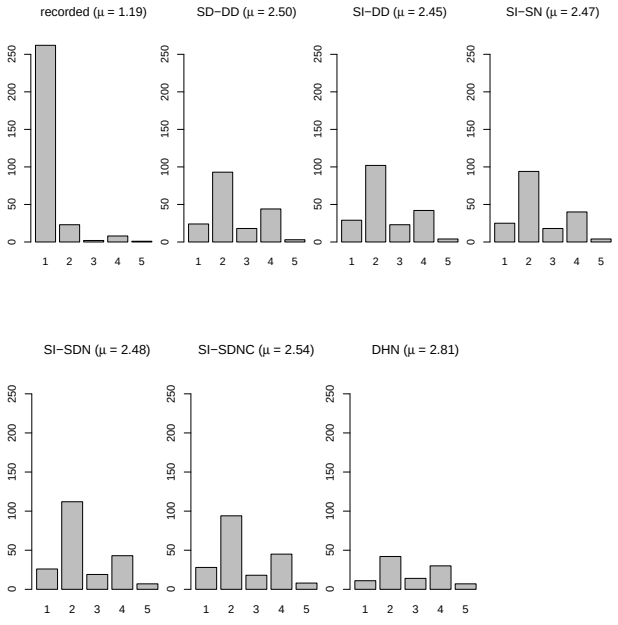


Figure 6: *Frequencies of similarity votes for each of the seven methods to the recorded reference sample as evaluated in the subjective listening test. 1 means very similar, 5 means very different.*

ods (Table 3). Concerning the adaptive methods this can be due to the small amount of speakers for some models, but for SI-SDN for example we had 10 pseudo-speakers in the average voice model. Furthermore improvements with adaptive modeling for Austrian German and Viennese were reported [5] with similar data sets.

Another reason could be that the phone overlap between the different varieties is not large enough for applying adaptive modeling directly. The larger phone overlap of 77% between Austrian German and Viennese supports this hypothesis.

For the varieties used for the full average voice, the phone set overlap was 26% (AT ∩ IVG ∩ GOI). For the variety pairs the phone set overlap was 33% (AT ∩ IVG, AT ∩ GOI) and 59% (GOI ∩ IVG). This suggests a pre-clustering of the data prior to training and adaptation, which is dependent on larger amounts of training data. It also shows that the distance between variety and standard in terms of phonetic overlap can be quite different for different varieties.

Concerning the overall similarity of synthesized samples to original ones we saw that we can achieve a satisfying modeling of overall similarity with all modeling methods except DHN (Figure 6). Assuming that overall similarity factors into variety similarity and speaker similarity, we can conclude that dialects and speakers can be modeled successfully.

Even if we see no significant differences between adaptive and speaker-dependent modeling with this data set, we would still favor the adaptive approach since it has shown its advantage in other experiments [5] and it does never decrease the quality (except for DHN). Furthermore the adaptive approach gives us additional possibilities for applications due to the common decision tree structure in the modeling of fast speech [15] or dialect interpolation [5] for example. The analysis of phone set overlaps points to a threshold that shows when it is possible to exploit the full potential of the adaptive approach.

6. Conclusion and future work

In this paper we have shown adaptive modeling methods for dialects. We have described our data selection and recording approach and have shown that speaker-dependent and adaptive approaches are able to model the overall similarity between synthetic and recorded speech. Although we found no significant differences between adaptive and speaker-dependent methods the adaptive approach is still beneficial for applications like fast speech and dialect interpolation. Furthermore we have built corpora for the Bad Goisern (GOI) and Innervillgraten (IVG) dialect and synthesized these dialects for the first time. These synthesizers can be applied in many fields like tourism and language learning. The corpora are an important step in dialect preservation.

In future work we want to include the (already available) data from female speakers and perform gender-dependent/independent modeling. Furthermore we want to investigate pre-clustering techniques that can be applied to small data sets.

It remains an open question how much overlap we need between varieties to fully exploit the adaptive approach ad how we should measure this overlap. As our GOI and IVG corpora have a larger phone set overlap with each other than either of them does with the AT corpus, building a combined average model of GOI and IVG could further assist the analysis. This model could then be compared with speaker-dependent models and dialect-dependent average models to further investigate the impact of phone set overlap on the final speech quality.

7. Acknowledgements

This research was funded by the Austrian Science Fund (FWF): P23821-N23. The Competence Center FTW Forschungszentrum Telekommunikation Wien GmbH is funded within the program COMET Competence Centers for Excellent Technologies by BMVIT, BMWA, and the City of Vienna. The COMET program is managed by the FFG.

8. References

[1] M. H. Cohen, J. P. Giangola, J. Balogh, Voice User Interface Design, Addison-Wesley, 2004.

[2] R. Dall, C. Veaux, J. Yamagishi, S. King, "Analysis of speaker clustering strategies for HMM-based speech synthesis", INTERSPEECH 2012.

[3] C. Wutiwiwatchai, A. Thangthai, A. Chotimongkol, C. Hansakunbuntheung, N. Thatphithakkul, "Accent level adjustment in bilingual Thai-English text-to-speech synthesis", ASRU 2011, 295-299.

[4] M. Wester, R. Karhila, "Speaker similarity evaluation of foreign-accented speech synthesis using HMM-based speaker adaptation", ICASSP 2011, 5372-5375.

[5] M. Pucher, D. Schabus, J. Yamagishi, F. Neubarth, V. Strom, "Modeling and interpolation of Austrian German and Viennese dialect in HMM-based speech synthesis", Speech Communication, 52(2):164-179, 2010.

[6] M. Toman, M. Pucher, "Structural KLD for cross-variety speaker adaptation in HMM-based speech synthesis", in Proc. SPPRA, Innsbruck, Austria, 2013.

[7] Y.-J. Wu, Y. Nankaku, and K. Tokuda, "State mapping based method for cross-lingual speaker adaptation in HMM-based speech synthesis", in Proc. INTERSPEECH, Brighton, United Kingdom, 2009, pp. 528-531.

[8] J. Yamagishi, O. Watts, "The CSTR/EMIME HTS system for Blizzard challenge 2010", in Blizzard Challenge Workshop, Kansai Science City, Japan, 2010.

[9] M. Pucher, N. Kerschhofer-Puhalo, D. Schabus, S. Moosmüller, G. Hofer, "Language resources for the adaptive speech synthesis of dialects", Proc. of SIDG 2012, Vienna, Austria.

[10] H Scheutz, "Deutsche Dialekte des Alpenraums", 2009, `http://www.argealp.org/atlas/data/atlas.html`.

[11] H. Scheutz, S. Aitzetmüller, Peter Mauser, "Drent und herent. Dialekte im salzburgisch-bayerischen Grenzgebiet. Mit einem sprechenden Dialektatlas auf CD-ROM", EuRegio Salzburg, 2007.

[12] J. Yamagishi, T. Kobayashi, "Average-voice-based speech synthesis using HSMM-based speaker adaptation and adaptive training", IEICE Trans. Inf. & Syst., vol. E90-D, no. 2, pp. 533543, Feb. 2007.

[13] K. Shinoda, T. Watanabe, "Acoustic modeling based on the MDL principle for speech recognition", Eurospeech-97, 99-102.

[14] J. Yamagishi, H. Zen, Y.-J. Wu, T. Toda, K. Tokuda, "The HTS-2008 System: Yet Another Evaluation of the Speaker-Adaptive HMM-based Speech Synthesis System", Proc. Blizzard Challenge 2008.

[15] M. Pucher, D. Schabus, J. Yamagishi, "Synthesis of fast speech with interpolation of adapted HSMMs and its evaluation by blind and sighted listeners". INTERSPEECH 2010, Makuhari, Japan, pp. 2186-2189.

Investigation of intra-speaker spectral parameter variation and its prediction towards improvement of spectral conversion metric

Tatsuo Inukai, Tomoki Toda, Graham Neubig, Sakriani Sakti, Satoshi Nakamura

Graduate School of Information Science, Nara Institute of Science and Technology,
tatsuo-i@is.naist.jp, tomoki@is.naist.jp,
neubig@is.naist.jp, ssakti@is.naist.jp, s-nakamura@is.naist.jp

Abstract

In spectral conversion of statistical voice conversion (VC), distance-based measures between the converted and target spectral parameters are often used as evaluation or training criteria. However, even if the same speaker utters the same sentence, the spectral parameters vary utterance by utterance, and thus, spectral distance between utterances still exists. Moreover, the original prosodic features of input speech are often kept unchanged in some VC systems, such those that function in real-time. In such cases, prosody of converted and target speech samples are different, and these differences increases spectral distance. These potential spectral variations are not considered in the conventional evaluation/training criterion. Thus, by constructing criteria that consider this spectral difference improvements in sound quality can be expected. In this paper, we investigate intra-speaker spectral variation between utterances of the same sentence. We also propose a method for predicting this variation from prosodic parameter differences between the corresponding utterances. We conduct experimental evaluations using many speech samples of the same sentence uttered by a single speaker, with results demonstrating that the proposed method effectively predicts the intra-speaker spectral variation from the observed prosodic changes.

Index Terms: voice conversion, training/evaluation criterion, intra-speaker spectral variation, prosodic differences, prediction

1. Introduction

Statistical voice conversion (VC) is an effective technique for modifying acoustic parameters to convert non-linguistic or para-linguistic information while keeping linguistic information unchanged [1]. It was originally proposed for speaker conversion to change the voice uttered by a source speaker as if it is uttered by a specific target speaker [2]. Recent progress in VC has achieved high-quality and real-time conversion [3]. These technologies can be used in various VC applications for augmenting human-to-human speech communication, such as speaking-aid for vocally handicapped people [4, 5], silent speech interfaces [6], bandwidth extension [7], and singing voice effectors [8]. Improving VC performance has the potential to contribute greatly to practical use of these applications.

In real-time VC systems for speaker conversion, short-term speech features, such as spectral parameters, are mainly converted with little delay using complex conversion functions. On the other hand, long-term speech features, such as F_0 patterns, are fundamentally difficult to convert in real-time. Therefore, simple conversion functions, such as a global linear transform, are often used to convert F_0 values frame by frame. Consequently, performance of the real-time VC system strongly depends on spectral conversion.

As for spectral conversion, various conversion or evaluation criteria have been proposed. One of the most standard criteria is a (weighted) distance measure between converted and target spectral parameters. It is used in the most widely used VC methods, with a Gaussian mixture models (GMM) based on minimum mean square error estimation [9] or maximum-likelihood estimation [10]. Some sophisticated model training methods using it as an optimization criterion have also been proposed [11, 12]. Recently the use of not only the distance-based criterion but also other criteria have been proposed. One of them is global variance (GV), which is the second order moment of a spectral parameter trajectory [10]. It has been reported that speech quality and conversion accuracy for speaker individuality in converted speech are significantly improved by considering both the distance-based criterion and the GV. It has also been reported that mutual information is also useful [13]. The effectiveness of these criteria has also been confirmed in model training [14, 15, 16]. These results suggest that it is useful to use additional criteria rather than only the distance-based criteria, although it causes a larger distance between converted and target spectral parameters (i.e., a larger conversion error).

Previous research has not carefully investigated how much spectral distance is acceptable in VC. By considering the amount of acceptable spectral distance, it may be possible to automatically determine weight parameter controlling the balance between the distance-based criterion and additional criteria. To clarify the acceptable distance, we focus on intra-speaker spectral variation, which is the spectral distance observed when the same speaker utters the same sentence many times. It is empirically known that intra-speaker spectral variation will not go to zero between utterances. Moreover, it has been reported that larger prosodic changes cause larger spectral differences [17]. Therefore, the acceptable spectral distance possibly changes according to prosodic differences between the converted and target voices.

In this paper, we investigate intra-speaker spectral variation using many speech samples of the same sentences uttered by a single speaker. Mel-cepstral distortion [18] is used as a metric to capture the intra-speaker spectral variation. Moreover, we propose a method to predict the intra-speaker spectral variation between two utterances from their differences of various prosodic parameters. This prediction is useful to determine the acceptable spectral distortion in each utterance-pair and it has a potential to develop better training, conversion, and evaluation metrics for spectral conversion in VC.

2. Basic Procedure of VC

In the statistical VC for speaker conversion, a parallel data set consisting of utterance pairs of the source and target speakers is used to train the conversion models for individual speech parameters. As the conversion model for spectral parameters, a conditional probability density function of the target speaker's spectral parameters given the source speaker's spectral parameters is often modeled by a GMM. On the other hand, as the conversion model for F_0 parameters, the following global linear transformation is often used:

$$\log \hat{F_0} = \frac{\sigma^{(t)}}{\sigma^{(s)}} \left(\log F_0 - \mu^{(s)} \right) + \mu^{(t)}, \quad (1)$$

where F_0 is the source speaker's F_0 value and $\hat{F_0}$ is the converted F_0 value. The conversion model parameters are $\mu^{(s)}$ and $\sigma^{(s)}$, which are mean and standard deviation values of log-scaled F_0 values of the source speaker, and $\mu^{(t)}$ and $\sigma^{(t)}$, which are those of the target speaker. Prosodic parameters, such as shape of F_0 pattern, phoneme duration, and power patterns, are kept unchanged in conversion. Note that it is also possible to convert them if real-time conversion processing is not necessary and linguistic contents are available. However, such a conversion is essentially difficult in real-time conversion processing without any linguistic contents.

In the conversion processing as mentioned above, the converted speech is generated by the converted spectral parameters and globally transformed F_0 values without any prosodic changes. Therefore, ideal converted spectral parameters will be spectral parameters of a speech sample uttered by the target speaker so that its prosody is the same as that of the source speaker. However, it is not straightforward to record such speech samples as in each utterance pair of the available parallel data the target speaker's prosody is usually different from the source speaker's prosody. Therefore, the target spectral parameters are not ideal ones. Nevertheless, in the traditional approach the spectral conversion model is basically trained so that the conversion error (i.e., the distance between the converted spectral parameters and the target spectral parameters) in the parallel data set is minimized.

3. Investigation of Intra-Speaker Spectral Parameter Variation

We investigate how much spectral parameters vary when a single speaker utters the same sentence and how much spectral parameters differ additionally by imitating prosody of other speakers.

3.1. Recording of speech samples

We recorded speech samples of the same sentence uttered by a single speaker. One Japanese male speaker uttered one sentence 200 times with his own prosody. He also uttered the same sentence while imitating prosody of other reference speakers. The number of reference speakers was 24 (12 male and 12 female). To make it easy to imitate the utterances, 1) analysis-synthesized speech samples were generated by converting F_0 values of speech samples of the reference speakers using **Eq.** (1) to make their F_0 ranges equivalent to that of the male speaker and 2) they were presented to the male speaker as reference speech samples during the recording. The male speaker recorded 8 utterances imitating each reference speaker's prosody. A total of 192 speech samples were recorded. The sampling frequency was 16 kHz.

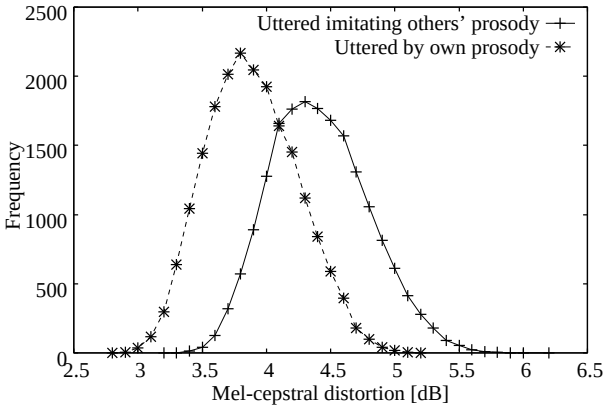


Figure 1: Frequency distribution of mel-cepstral distortion between utterances of the same sentence uttered by the same speaker.

3.2. Intra-speaker spectral parameter variation

The 1^{st} through 24^{th} mel-cepstral coefficients extracted by STRAIGHT analysis [19] were used as spectral parameters. Frame shift was 5 ms. Mel-cepstral distortion was calculated by performing dynamic time warping (DTW) in each utterance pair.

Figure 1 shows the frequency distribution of the mel-cepstral distortion for all utterance-pairs, one is a frequency distribution for speech samples with the male speaker's own prosody and the other is that for speech samples with the different speakers' prosody. We can see that even in the same sentence with the same speaker, the mel-cepstral distortion is not 0. For the speech samples with the speaker's own prosody, the mean value is 3.9 dB and the standard deviation value is 0.35 dB. On the other hand, for the speech samples with the different speakers' prosody, the spectral variation tends to be larger; its mean value is 4.4 dB and its standard deviation value is 0.38 dB.

These results suggest that 1) it is not necessary to decrease mel-cepstral distortion to 0 in VC and 2) as prosodic differences between the source and target speakers are larger, a larger mel-cepstral distortion will be acceptable.

4. Prediction of Spectral Parameter Variation

While the source and target speakers' prosody is usually different from each other in available parallel data sets, it is ideal to predict target spectral parameters in each utterance-pair when the target speaker imitates prosody of the source speaker. However, this is not straightforward to do. Although a method for predicting spectral parameter changes according to F_0 changes has been proposed [17], it still needs training data consisting of many speech samples of the same linguistic contents uttered by the target speaker with different F_0 values. It is laborious work to additionally record such a data set. Therefore, we simplify the problem to be solved. We predict spectral distortion between the original speech sample of the target speaker in the parallel data and a practically unavailable speech sample uttered by the target speaker while imitating prosody of the corresponding utterance of the source speaker. Namely, we predict not an unobserved spectral feature vector itself but its distance from an observed spectral feature vector. The predicted spectral dis-

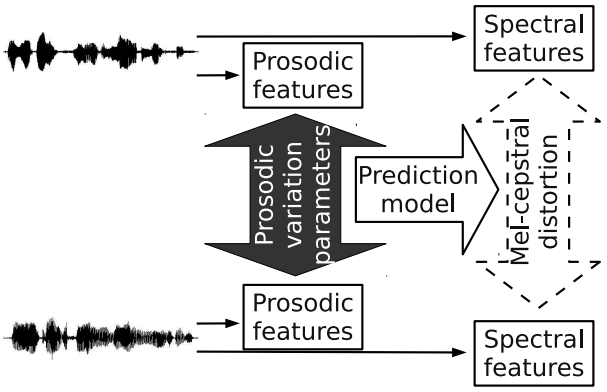


Figure 2: Prediction procedure of mel-cepstral distortion from prosodic variation parameters.

tortion is still useful as it shows acceptable spectral distortion depending on prosodic differences in each utterance-pair.

As a first step to achieve such a prediction, in this paper we propose a method for predicting the mel-cepstral distortion from prosodic variation parameters capturing prosodic differences using many speech samples of the same sentence uttered by a single speaker. **Figure 2** shows the prediction procedure. Prosodic parameters extracted from an utterance pair and the prosodic variation parameters are calculated as an explanation variable. The mel-cepstral distortion is also calculated in this utterance-pair as a target variable. Then, the mel-cepstral distortion is predicted from the prosodic variation parameters. In a practical application, the prosodic variation parameters are calculated between the source and target speech samples in each utterance pair for training or evaluation. Finally, the mel-cepstral distortion is predicted from these samples. The predicted mel-cepstral distortion is regarded as an acceptable distortion between the converted and target spectral parameters. Note that this distortion varies utterance by utterance. It is inevitable to develop a sentence/speaker-independent prediction model to make it possible to apply this prediction model in practical VC conditions.

4.1. Prediction Model

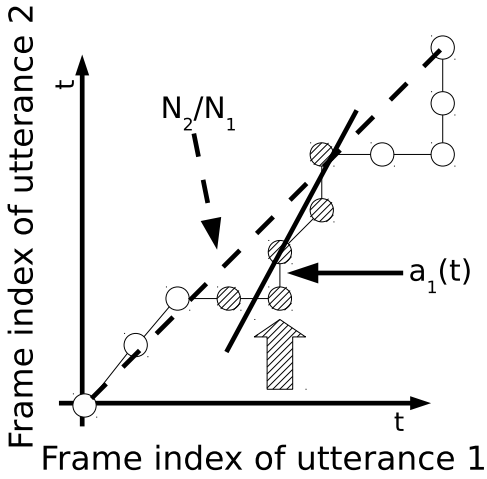
A multiple linear regression model is used to predict the mel-cepstral distortion from the prosodic variation parameters as follows:

$$\hat{m}_{i,j} = \boldsymbol{a}^{\top} \boldsymbol{p}_{i,j} + c, \tag{2}$$

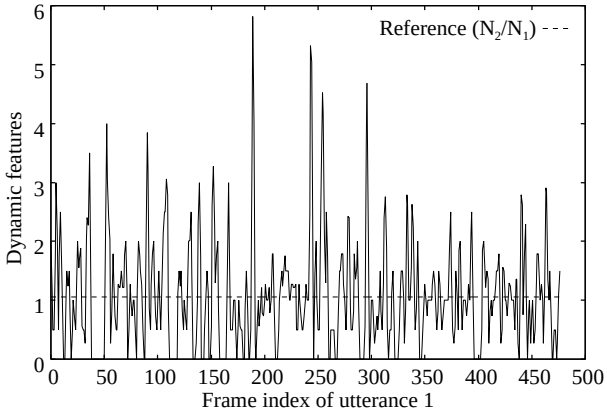
where $\hat{m}_{i,j}$ is the predicted mel-cepstral distortion between the i^{th} utterance and the j^{th} utterance, $\boldsymbol{p}_{i,j}$ is a prosodic variation parameter vector between these utterances, $\boldsymbol{a}$ is a regression coefficient vector, and c is a bias value. The regression coefficient vector and the bias value are determined by the least square error estimation. In this paper, the mel-cepstral distortion and the prosodic variation parameters are calculated using only voice active frames, which are automatically extracted with normalized waveform power.

4.2. Prosodic Variation Parameters

Several prosodic variation parameters are used in the prediction. Duration distortion and DTW distortion capture the difference in duration. Voiced/unvoiced error rate and F_0 distortion capture the difference in F_0 patterns. Power distortion captures the difference in power patterns. These parameters take positive values and only take zeros when prosody of two speech samples is completely the same as each other.



(a) Time warping function



(b) Dynamic feature sequence of time warping function

Figure 3: Calculation of DTW distortion.

4.2.1. Duration Distortion

To capture a difference of total duration over an utterance, duration distortion is calculated as follows:

$$D_{\text{dur}} = \log N_l - \log N_s, \tag{3}$$

where N_l is the number of frames extracted from a longer utterance, and N_s is the number of frames extracted from a shorter utterance.

4.2.2. DTW Distortion

To capture the difference in local duration, DTW distortion is calculated as shown in **Figure 3**. First, temporally dynamic features of the time warping function are determined by DTW, which is given by the slope of a regression line as shown in $a_1(t)$ in **Figure 3(a)**, and calculated at each frame of each utterance. One example of the dynamic feature sequence over an utterance is shown in **Figure 3(b)**. If there is no difference in local duration, the time warping function is represented as a line and the slope at every frame is equivalent to its constant slope N_2/N_1 as shown in **Figure 3(a)**. The DTW distortion is calculated as a difference between the dynamic features and the

constant slope as follows:

$$D_{\text{DTW}} = \frac{1}{2N_1}\sqrt{\sum_{t=1}^{N_1}\left(a_1(t) - \frac{N_2}{N_1}\right)^2} + \frac{1}{2N_2}\sqrt{\sum_{t=1}^{N_2}\left(a_2(t) - \frac{N_1}{N_2}\right)^2}, \quad (4)$$

where N_1 and N_2 are the number of frames of utterance 1 and utterance 2, and $a_1(t)$ and $a_2(t)$ are the dynamic feature at frame t over the utterance 1 and utterance 2. All frame pairs from frame $t-1$ to $t+1$ over the utterance 1 are used to fit a regression line to calculate $a_1(t)$. In a similar way, $a_2(t)$ is also calculated.

4.2.3. Voiced/Unvoiced Error Rate

To capture the difference of voiced/unvoiced frames, a voiced/unvoiced error rate between frames time-aligned by DTW is calculated as follows:

$$D_{\text{U/V}} = \frac{1}{N}\sum_{t=1}^{N} e(t), \quad (5)$$

where N is the number of time-aligned frame pairs, $e(t)$ is a function that returns 0 when voice/unvoiced information is the same at frame-pair t and returns 1 when they are different.

4.2.4. F_0 Distortion

To capture the difference of F_0 patterns, F_0 distortion is calculated between time-aligned frames by DTW as follows:

$$D_{F_0} = \frac{1}{N_v}\sqrt{\sum_{t=1}^{N_v}\left(\log(F_0^{(1)}(t)) - \log\left(F_0^{(2)}(t)\right)\right)^2}, \quad (6)$$

where N_v is the number of voiced frame pairs, and $F_0^{(1)}(t)$ and $F_0^{(2)}(t)$ are F_0 values of individual utterances at frame pair t.

We also calculate a maximum value ($D_{F_0}^{(\text{max})}$) and a minimum value ($D_{F_0}^{(\text{min})}$) of the absolute difference of log-scaled F_0 in each utterance pair.

4.2.5. Power Distortion

To capture the difference of power patterns, power distortion is calculated between time-aligned frames by DTW as follows:

$$D_{\text{pow}} = \frac{1}{N}\sqrt{\sum_{t=1}^{N}\left(p^{(1)}(t) - p^{(2)}(t)\right)^2}, \quad (7)$$

where N is number of frame pairs, $p^{(1)}(t)$ and $p^{(2)}(t)$ are normalized power values of individual utterances at frame pair t.

We also calculate a maximum value ($D_{\text{pow}}^{(\text{max})}$) and a minimum value ($D_{\text{pow}}^{(\text{min})}$) of the absolute difference of the normalized power in each utterance pair.

4.3. Normalization of Speaker-Dependency

The prosodic variation parameters and the mel-cepstral distortion are affected by speaker individuality. To reduce the impact of speaker dependence on these parameters, all parameters are normalized so that their mean and standard deviation values are equal to 0 and 1 in each speaker.

This normalization can be straightforwardly applied to the prosodic variation parameters (i.e., variable calculable from the input) using the parallel data in practical VC conditions. On the other hand, it is not straightforward to apply it to the mel-cepstral distortion (i.e., a target variable). Namely, the normalized mel-cepstral distortion is predicted but the unnormalized mel-cepstral distortion is hard to predict. Nevertheless, the normalized mel-cepstral distortion is still effective to improve the conventional training, conversion, and evaluation criteria because it captures additional information about the acceptable spectral distortion varying utterance by utterance.

Table 1: Prediction results of mel-cepstral distortions.

Regression model		Correlation coefficient
Speaker-dependent	Sentence-dependent	0.76
Speaker-dependent	Sentence-independent	0.75
Speaker-independent	Sentence-independent without normalization	0.64
Speaker-independent	Sentence-independent with normalization	0.72

5. Experiments

5.1. Experimental Conditions

We recorded speech data of 5 speakers (4 males, 1 female) in the same way as described in **Section 3.1**. Male 1 uttered 6 sentences 200 times, and the other speakers uttered 4 sentences 50 times. These sentences were extracted from the ATR Japanese speech database [20]. The 1^{st} through 24^{th} mel-cepstral coefficients extracted by STRAIGHT analysis [19] were used as spectral parameters. F_0 values were extracted by the F_0 estimation method of STRAIGHT analysis [21]. The sampling frequency was 16 kHz. Frame shift was 5 ms.

To evaluate prediction accuracy, we calculated a correlation coefficient between the predicted mel-cepstral distortion and the observed mel-cepstral distortion. We evaluated the following four models:

1) speaker- and sentence-dependent models: a single prediction model was trained and evaluated for each speaker and each sentence,

2) speaker-dependent and sentence-independent models: a single prediction model was trained and evaluated for each speaker using all of his/her sentences,

3) speaker- and sentence-independent models without normalization: a global prediction model was trained for all speakers using their all sentences without normalization described in **Section 4.3**,

4) speaker- and sentence-independent models with normalization: a global prediction model was trained for all speakers using their all sentences with the normalization.

In each case, five-fold cross validation was employed. All combinations of utterance-pairs of the same speaker and the same sentence were considered. In the speaker-independent model, the number of utterances of Male 1 was reduced to the same number of utterances of the other speakers. We also evaluated the effect of individual prosodic variation parameters on prediction accuracy by adding them as explanatory features one by one in the speaker- and sentence-dependent model.

Table 2: Correlation coefficients between individual prosodic difference parameters.

–	D_{DTW}	$D_{U/V}$	D_{F_0}	$D_{F_0}^{max}$	$D_{F_0}^{min}$	D_{pow}	D_{pow}^{max}	D_{pow}^{min}
D_{dur}	0.22	0.05	0.10	0.02	0.03	0.11	0.04	0.03
D_{DTW}	–	0.25	0.29	0.23	0.05	0.27	0.16	0.04
$D_{U/V}$	–	–	0.40	0.24	0.20	0.58	0.32	0.12
D_{F_0}	–	–	–	0.65	0.28	0.37	0.19	0.12
$D_{F_0}^{max}$	–	–	–	–	0.09	0.26	0.17	0.07
$D_{F_0}^{min}$	–	–	–	–	–	0.14	0.06	0.04
D_{pow}	–	–	–	–	–	–	0.63	0.20
D_{pow}^{max}	–	–	–	–	–	–	–	0.10

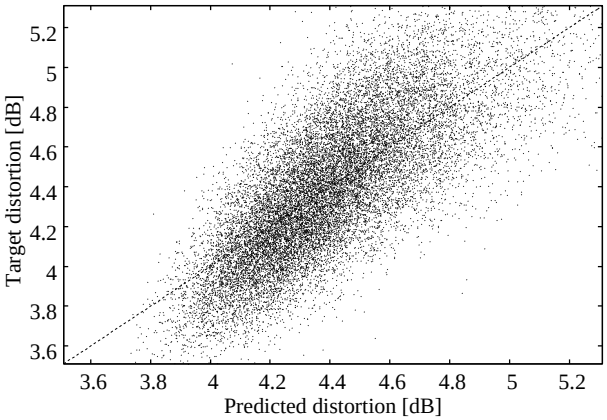


Figure 4: Scatter diagram of target mel-cepstral distortion and that predicted by sentence-dependent model (for Male 1).

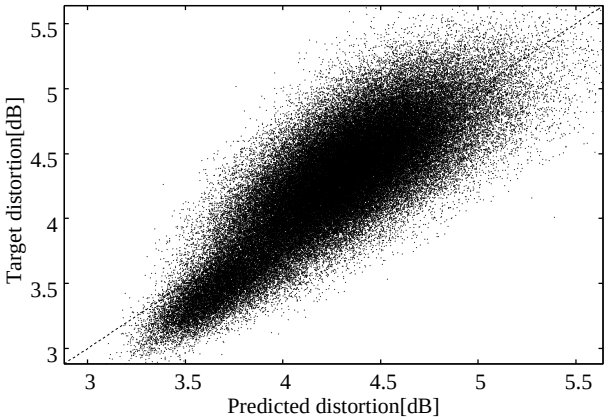


Figure 5: Scatter diagram of target mel-cepstral distortion and that predicted by sentence-independent model (for Male 1).

5.2. Experimental results

Table 1 shows results of individual prediction models using all prosodic variation parameters. The speaker- and sentence-dependent model is capable of predicting the mel-cepstral distortion with accuracy of 0.76 in correlation coefficient. A scatter plot between the predicted mel-cepstral distortion and the observed mel-cepstral distortion is shown in **Figure 4**. We can see a tendency that the prediction error is smaller as the observed mel-cepstral distortion is also smaller. The speaker-dependent and sentence-independent model does not cause any adverse effects and its correlation coefficient is 0.75 as shown in **Table 1**. Its scatter plot is shown in **Figure 5**. From this we can see that the proposed method is not sensitive to a change of sentence content. **Table 1** also shows results of the speaker-independent models with/without the normalization. If the normalization is not performed, the correlation coefficient decreases to 0.64. This result shows that the prediction model is strongly affected by the speaker differences. We can also observe this degradation in a scatter plot as shown in **Figure 6**. This degradation is alleviated by using normalization as shown in **Figure 7**.

Table 2 shows correlation coefficients between each prosodic variation parameter pairs. We can see that correlation coefficients tend to be low except for F_0 and power distortion and their maximum values. Therefore, each of the other prosodic variation parameters represents different property of prosodic differences. **Figure 8** shows changes of the correlation coefficient by adding the prosodic variation parameters to the feature set one by one. The DTW distortion has a great contribution to the prediction for all speakers. By further adding only the F_0 distortion and the power distortion, the prediction accuracy becomes almost equivalent to that achieved by using all prosodic variation parameters.

These results suggest that 1) the mel-cepstral distortion can be predicted using only three prosodic parameters (the DTW distortion, the F_0 distortion, and the power distortion) and 2) the speaker- and sentence-independent prediction model can be trained using normalization of speaker differences in each parameter.

6. Conclusions

In this paper, we investigated intra-speaker spectral variation between utterances of the same sentence. It was found that larger prosodic differences cause larger spectral variations, and acceptable spectral distortion in VC varies by prosodic variation. To predict the spectral variations caused by the prosodic differences, we proposed a prediction method using a multiple linear regression model to predict the mel-cepstral distortion from several prosodic variation parameters. The experimental results have demonstrated that 1) the mel-cepstral distortion is predicted relatively well by the proposed method (the correlation coefficient is more than 0.7), 2) the prediction model is robust against sentence differences, and 3) the prediction model is sensitive to the speaker differences but this issue is well alleviated by the parameter normalization, and 4) good prediction accuracy is achieved using only three prosodic parameters. We plan to construct training, conversion, and evaluation metrics considering the predicted spectral variation.

7. Acknowledgements

Part of this work was supported by JSPS KAKENHI Grant Number 22680016.

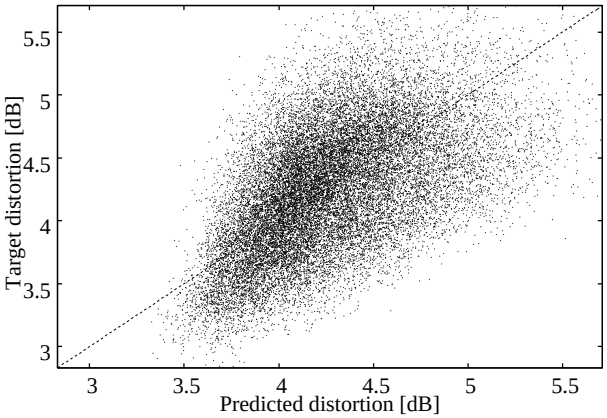


Figure 6: Scatter diagram of target mel-cepstral distortion and that predicted by speaker-independent model (for all speakers).

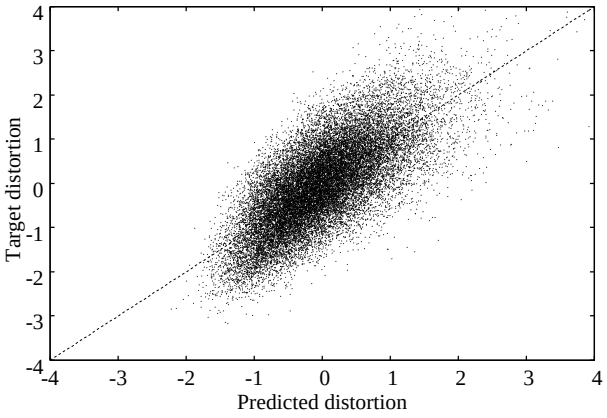


Figure 7: Scatter diagram of target mel-cepstral distortion and that predicted by normalized speaker-independent model (for all speakers).

8. References

[1] Y. Stylianou, "Voice transformation: a survey," *Proc. ICASSP*, pp. 3585–3588, Apr. 2009.

[2] M. Abe, S. Nakamura, K. Shikano, and H. Kuwabara, "Voice conversion through vector quantization," *J. Acoust. Soc. Jpn.* (E), Vol. 11, No. 2, pp. 71–76, 1990.

[3] T. Toda, T. Muramatsu, H. Banno, "Implementation of computationally efficient real-time voice conversion," *Proc. of INTERSPEECH*, Sept. 2012.

[4] Kain. A. B., Hosom. J. P., Niu. X., van Santen. J. P., Fried-Oken. M., and Staehely. J, "Improving the intelligibility of dysarthric speech," *Speech communication*, Vol. 49, No. 9, pp.743–759, 2007.

[5] H. Doi, K. Nakamura, T. Toda, H. Saruwatari, and K. Shikano, "Esophageal speech enhancement based on statical voice conversion with Gaussian mixture models.", *IEEE Trans. Inf. & Syst.*, Vol. E93-D, No. 9, pp. 2472–2482, 2010.

[6] T. Toda, M. Nakagiri, K. Shikano, "Statistical voice conversion techniques for body-conducted unvoiced speech enhancement," *IEEE Trans. ASLP*, vol. 20, No. 9, pp. 2505–2517, Nov. 2012.

[7] P. Jax and P. Vary. "On artificial bandwidth extension of telephone speech." *Signal Processing*, Vol. 83, pp. 1707–1719, 2003.

[8] H. Doi, T. Toda, T. Nakano, M. Goto, and S. Nakamura, "Singing Voice Conversion Method Based on Many-to-Many Eigenvoice Conversion and Training Data Generation Using a Singing-to-Singing Synthesis System." *APSIPA Annual Summit and Conference*, 2012.

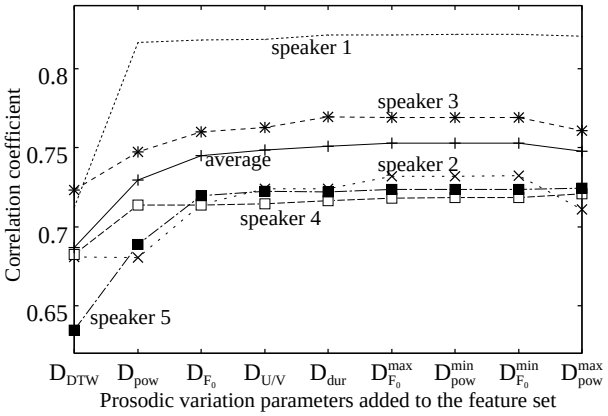


Figure 8: Correlation coefficient when adding prosodic variation parameters one-by-one.

[9] Y. Stylianou, O. Cappe, and E. Moulines. "Continuous probabilistic transform for voice conversion." *IEEE Trans. SAP*, Vol. 6, No. 2, pp. 131–142, 1998.

[10] T. Toda, A.W. Black, and K. Tokuda. "Voice conversion based on maximum likelihood estimation of spectral parameter trajectory." *IEEE Trans. ASLP*, Vol. 15, No. 8, pp. 2222–2235, 2007.

[11] Yi-Jian Wu and Ren-Hua Wang, "Minimum Generation Error Training for HMM-Based Speech Synthesis." *in Proc. ICASSP*, pp. 89–92, 2006.

[12] H. Zen, Y. Nankaku, and K. Tokuda. "Continuous stochastic feature mapping based on trajectory HMMs." *IEEE Trans. ASLP*, Vol. 19, No. 2, pp. 417–430, 2011.

[13] Hsin-Te Hwang, Yu Tsao, Hsin-Min Wang, Yih-Ru Wang and Sin-Horng Chen "A Study of Mutual Information for GMM-Based Spectral Conversion." *in Proc. Interspeech*, 2012.

[14] H. Benisty and D. Malah, "Voice Conversion using GMM with Enhanced Global Variance." *in Proc. Interspeech*, pp. 669–672, 2011.

[15] Zen. H, Gales. M. J F, Nankaku. Y and Tokuda. K, "Product of Experts for Statistical Parametric Speech Synthesis," *Audio, Speech, and Language Processing, IEEE Transactions on*, Vol.20, No.3, pp.794–805, 2012

[16] Hwang. H. T., Tsao. Y., Wang. H. M., Wang. Y. R., and Chen. S. H, "Exploring mutual information for GMM-based spectral conversion." *In Chinese Spoken Language Processing (ISCSLP) 2012 8th International Symposium on*, pp. 50–54, 2012.

[17] N. Minematsu and S. Nakagawa, "Analysis and modeling of spectral variations caused by F_0 changes" *Acoust. Soc. Jpn.*, Vol. 55, No. 3, pp. 165–174, 1999. (In Japanese).

[18] R. Kubichek, "Mel-cepstral distance measure for objective speech quality assessment." *Communications, Computers and Signal Processing, 1993., IEEE Pacific Rim Conference on.* Vol. 1, pp. 125–128, 1993.

[19] H. Kawahara, I. Masuda-Katsuse and A. deCheveigné, "Restructuring speech representations using a pitch-adaptive time- frequency smoothing and an instantaneousfrequency-based F_0 extraction: Possible role of a repetitive structure in sounds," *Speech Communication.*, Vol. 27, No. 3–4, pp. 187–207, 1999.

[20] Y. Sagisaka, K. Takeda, M. Abe, S. Katagiri, T. Umeda and H. Kuwahara, "A large-scale Japanese speech database.", ICSLP90, pp.1089–1092, 1990.

[21] H. Kawahara, H. Katayose, A. deCheveigné, and R.D. Pateterson, "Fixed point analysis of frequency to instantaneous frequency mapping for accurate estimation of f0 and periodicity," *Proc. Eurospeech*, Vol. 99, pp.2781–2784, 1999.

Text to Speech in New Languages without a Standardized Orthography

Sunayana Sitaram, Gopala Krishna Anumanchipalli, Justin Chiu, Alok Parlikar and Alan W Black

Language Technologies Institute, Carnegie Mellon University

{ssitaram, gopalakr, jchiu1, aup, awb}@cs.cmu.edu

Abstract

Many spoken languages do not have a standardized writing system. Building text to speech voices for them, without accurate transcripts of speech data is difficult. Our language independent method to bootstrap synthetic voices using only speech data relies upon cross-lingual phonetic decoding of speech. In this paper, we describe novel additions to our bootstrapping method. We present results on eight different languages---English, Dari, Pashto, Iraqi, Thai, Konkani, Inupiaq and Ojibwe, from different language families and show that our phonetic voices can be made understandable with as little as an hour of speech data that never had transcriptions, and without many resources in the target language available. We also present purely acoustic techniques that can help induce syllable and word level information that can further improve the intelligibility of these voices.

Index Terms: speech synthesis, synthesis without text, languages without an orthography

1. Introduction

Recent developments in speech and language technologies have revolutionized the ways in which we access information. Advances in speech recognition, speech synthesis and dialog modeling have brought out interactive agents that people can talk to naturally and ask for information. There is a lot of interest in building such systems especially in multilingual environments. Building speech and language systems typically requires significant amounts of data and linguistic resources. For many spoken languages of the world, finding large corpora or linguistic resources is difficult. Yet, these languages have many native speakers around the world and it would be very interesting to deploy speech technologies in them.

Our work is about building text-to-speech systems for languages that are purely spoken languages: they do not have a standardized writing system. These languages could be mainstream languages such as Konkani (a western Indian language with over 8 million speakers), or dialects of a major language that are phonetically quite distinct from the closest major language. Building a TTS system usually requires training data consisting of a speech corpus with corresponding transcripts. However, for these languages that aren't written down in a standard manner, one can only find speech corpora. Our current efforts focus on building speech synthesis systems when our training data doesn't contain text.

It may seem futile to build a TTS system when the language at hand doesn't have a text form. Indeed, if there is no text at training time, there won't be text at test time, and then one might wonder why we need a TTS system at all. However, consider the use case of deploying a speech-to-speech translation of video lectures from English into Konkani. We have to synthesize speech in this "un-written" language from the output of a machine translation system. Even if the language at hand may not have a text form, we need some intermediate representation that can act as a text form that the machine translation system can produce. A first approximation of such a form is phonetic strings. Another use case for which we need TTS without text is, say, deploying a bus information system in Konkani. Our dialog system could have information about when the next bus is, but it has to generate speech to deliver this information. Again, one can imagine using a phonetic form to represent the speech to be generated, and produce a string of phones from the natural language generation model in the bus information dialog system.

The work we present here is our continued effort in improving text to speech for languages that do not have a standardized orthography. We have built voices for several languages, from purely speech corpora, and produced understandable synthesis. We use cross-lingual phonetic speech recognition methods to do so. Phone strings are not ideal for TTS, however, as a lot of information is contained in higher level phonological units including the syllables and words that can help produce natural prosody. However, detecting words from speech corpus alone is a difficult task. We have explored how purely acoustic techniques can be used to detect word like units in our training speech corpus and use this to further improve the intelligibility of speech synthesis.

2. Relation to prior work

Speech to speech translation typically involves a cascade of three models: an automatic speech recognition system (ASR) in the source language, a statistical machine translation system (SMT), and a text to speech engine (TTS) in the target language. Generally, these three models are developed independently of each other. Recent work such as [1, 2, 3, 4] has looked into deeper integration of this pipeline, but the general assumption here is that the target language has an orthography.

If the target language of speech to speech translation does not have a written form, it has been proposed that one be defined, though training people to use it consistently is in itself very hard and prone to inconsistencies (e.g. Iraqi Arabic transcription techniques in the recent TRANSTAC Speech to Speech Translation Project, see [5]). Our proposal is to use a phonetic-like representation of the target speech, derived acoustically as the orthography to use. [5, 6] have investigated such an approach.

Changes have been proposed to SMT modeling methods [7, 8] to specifically deal with phoneme strings in the target language. In order to induce the automatic phonetic writing form, we use an ASR system in a foreign language and

adapt the acoustic model to match the target speech corpus. Speech synthesis voices are typically built from less data compared to speech recognition systems. Acoustic model adaptation with limited resources can be challenging [9]. [10] has recently proposed a rapid acoustic model adaptation technique using cross-lingual bootstrapping that showed improvements in the ASR of under-resourced languages. Our model adaptation technique is somewhat similar to that method, but we optimize the adaptation towards better speech synthesis, and have only acoustic data in the target language.

In preliminary work in this direction [11] we proposed a method to devise a writing system. We also proposed using existing techniques to automatically induce words and syllables from a string of phonemes [12]. In this work, we propose using acoustic information to derive higher level phonological units, which is language independent and more reliable than inducing structures using noisy ASR output.

Although such representations may be difficult for a native speaker to write, an SMT system can help bridge the gap from a source language to the target phonetic representation of the language. [13] models pronunciation variability based on articulatory features and is more suited for our purpose (since ASR transcript could be noisy) and we plan to use such models in the future.

3. Data and resources

We used audio from eight languages from four diverse language families for this research. Our audio data ranged from almost two hours of speech to less than six minutes, as shown in Table 1.

Language	Size (minutes)
English	111
Dari	52
Iraqi	62
Pashto	39
Thai	25
Ojibwe	12
Inupiaq	5.5
Konkani	5.5

Table 1: Audio data sizes

Our English data was from the Blizzard Challenge [14] 2013 audio book task, recorded by a professional voice recording artist.

Dari is a dialect of Persian that is used in Afghanistan as an official language and also spoken in parts of Iran and Tajikistan. It has over 18 million native speakers. Pashto is a also an official language of Afghanistan and has over 40 million speakers. The Dari and Pashto corpora are from the DARPA TRANSTAC project. Iraqi Arabic is a dialect of Arabic spoken in Iraq and has about 15 million speakers. The Iraqi Arabic corpora were provided by BBN as part of the DARPA BOLT project.

The Thai language is spoken by over 20 million people and is the official language of Thailand. We used the Thai speech corpora from the SPICE [15] dataset.

Inupiaq is an Inuit language spoken by about 2100 people in northern and northwestern Alaska. Ojibwe is spoken in Canada and the United States and has around 56000 native speakers. Both Inupiaq and Ojibwe use the Latin script in their written forms. Our data for Inupiaq and Ojibwe came from a corpus collected as part of the Endangered Languages project at Carnegie Mellon University.

Konkani is an official language of India and is used primarily in Goa and Karnataka. It has over 8 million native speakers. Konkani does not have its own script, and native Konkani speakers use Devanagari, Latin, Kannada, Malayalam and even Arabic scripts to write it. We used a corpus of Konkani from the CMU SPICE project [15].

We used the CMU Sphinx [16] speech recognition toolkit in allphone mode as our phonetic decoder and to train new acoustic models. We used the Festvox voice building tools to build CLUSTERGEN [17] voices for the Festival [18] speech synthesizer. CLUSTERGEN is a type of statistical parametric synthesizer that is more robust to noise than other methods such as unit selection. Our method can be used with any parametric synthesis technique.

Our phonetic decoder used trigram phonetic language models built from German and Marathi data. For the German language model, we used the Europarl [19] corpus and for the Marathi language model, we used a corpus created by collecting news stories from a Marathi news website, Esakal. The words are expanded to their phonetic forms using statistically trained letter to sound rules in the respective language. We used a single acoustic model, the Wall Street Journal (WSJ) English acoustic model provided with CMU Sphinx.

We used the TestVox[20] tool to run listening tests online.

4. Overview of our approach

Figure 1 shows a block diagram of the components and flow of our approach.

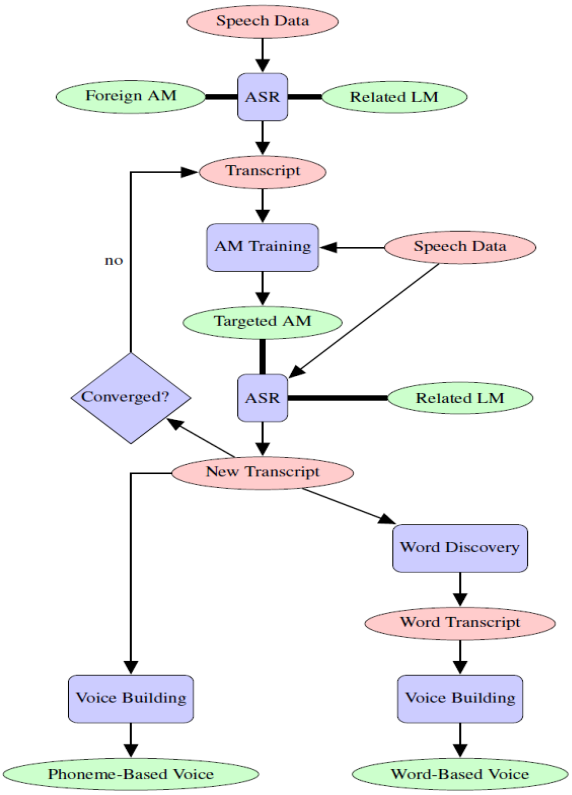


Figure 1: Overview of our approach

First, we decode the audio in the target language with a phonetic decoder using an acoustic model and language model from another language. Then, using the transcripts obtained from decoding and the speech corpus, we iteratively build new targeted acoustic models until convergence. We use the phonetic transcripts to build synthetic voices and evaluate them objectively using the Mel-Cepstral Distance (MCD) [21] and subjectively using human listening tasks. We also automatically induce syllable and word-like structures on these transcripts and build syllable and word based synthetic voices. The next few sections describe these steps in greater detail.

5. Bootstrapping synthetic voices

For phonetic decoding, we use an acoustic model and phoneme language model from a related language. For our experiments in this paper, we used the English WSJ acoustic model for decoding all languages. Using the WSJ acoustic model for decoding English speech is not fair, but we used it to keep the acoustic model consistent in all our experiments. Ideally for phonetic decoding, an acoustic model and phoneme language model from a closely related language are more appropriate. To simulate this in our experiments, we used Marathi and German phonetic language models, as listed in Table 2.

Language	Acoustic Model	Language Model
English	WSJ	German
Dari	WSJ	Marathi
Iraqi	WSJ	German
Pashto	WSJ	Marathi
Thai	WSJ	Marathi
Ojibwe	WSJ	German
Inupiaq	WSJ	German
Konkani	WSJ	Marathi

Table 2: Acoustic and Language models used for cross lingual decoding

After decoding speech in the target language using the appropriate acoustic and language models, we iteratively train new acoustic models using the decoded transcript as the text and the original audio as the speech. At each stage of the iterative process, we calculate the MCD of the voice built using the decoded transcript from that iteration. Figure 3, 4 and 5 show the MCD of voices built using these transcripts for various languages.

Figure 2 shows the MCDs of transcripts obtained on English, Dari and Iraqi Arabic. English has about two hours of speech while Dari and Iraqi Arabic have about one hour of speech. We see that there is a big drop in MCD value from the first iteration to the second, in which the targeted acoustic model is built. In the case of English, iteration 7 has the lowest MCD, after which it rises slightly. For Iraqi Arabic and Dari, the MCD continues to fall until the last iteration.

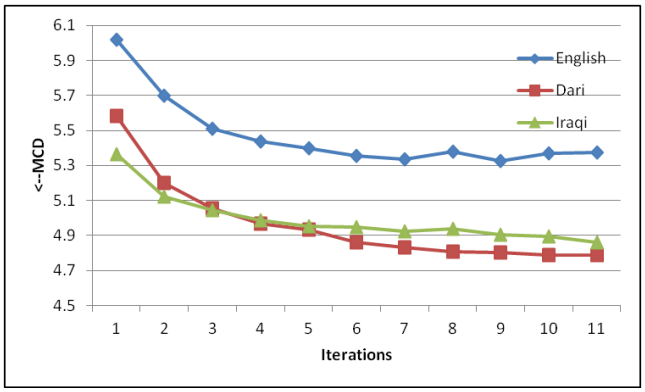


Figure 2: Iterative targeted Acoustic Models for languages with ~1 hour of speech

Figure 3 shows the MCD graph for Pashto and Thai, both of which have around 30 minutes of speech. We see that the MCD for Pashto in the first three iterations falls rapidly and then does not change much, while for Thai, there is a big drop after the first iteration, which is consistent with the results for English, Dari and Iraqi Arabic. There is a large rise in MCD at iteration 7 for Thai, but it falls again in the next iteration. We can see that even with half an hour of speech, our iterative method produces better transcripts than the base decoding with the WSJ acoustic model.

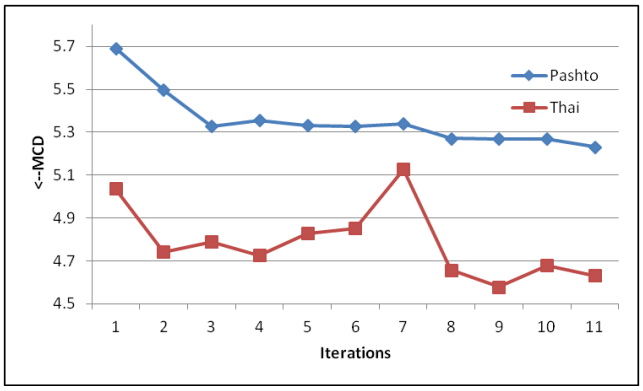


Figure 3: Iterative targeted Acoustic Models for languages with ~30 minutes of speech

Figure 4 shows results for Ojibwe, which has 12 minutes of speech and Inupiaq and Konkani, both of which have around five minutes of speech. We see that for Ojibwe, the MCD rises slightly after the first iteration and then falls after the fifth iteration, with the difference in the MCD between the base and best iteration being 1.43. This shows that even with just 12 minutes of speech, the iterative method is able to come up with a better transcript than just the base decoding. However, for both Inupiaq and Konkani, we see that the MCD rises after the first iteration. This is probably because of the phonetic complexity of these languages and the amount of speech is too small to build even targeted acoustic models.

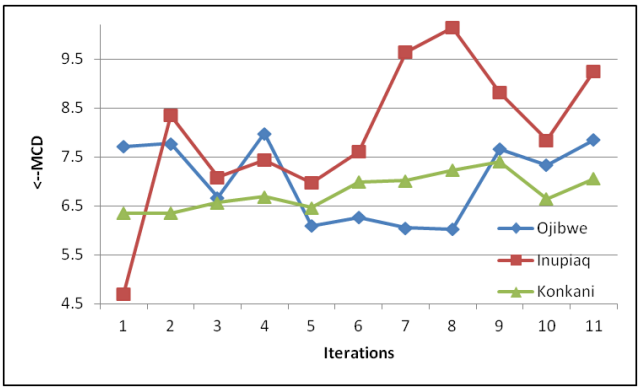


Figure 4: Iterative targeted Acoustic Models for languages with < 15 minutes of speech

Overall, we see that with a reasonable amount of speech data, the iterative targeted acoustic models produce better phoneme transcripts than just using base decoding from a cross lingual phonetic decoder, shown here on a variety of languages.

Throughout the iterations, we kept the language model used by the ASR consistent. One obvious extension of this approach is to adapt the language model at each iteration. However, preliminary experiments on interpolating the original language model at each iteration with the new transcript did not yield improvements in MCD.

6. Improved synthesis with syllables and words

So far, we have discussed the bootstrapping method which produces phoneme transcripts of the audio, which may be noisy. However, Text to Speech systems typically benefit from using syllable and word level information. So, we try to automatically induce syllables and word-like units from the phoneme transcripts.

We syllabified English and Dari transcripts with the lowest MCD in the 10 iterations. To obtain syllables, we use heuristic rules built into the Festival speech synthesizer to join phonemes in the transcripts. We treated the syllables as words and added appropriate entries in the lexicon.

For inducing word-like units, we used cross-lingual information to train a Conditional Random Field (CRF) model. We created training data for the CRF by extracting phonemes and word boundaries from the German Europarl data. We used CRF++ [22] to train a German model that could group phoneme sequences into world-like units and used the same English and Dari transcripts used for syllabification earlier to test the model. We discarded words that were rare (< 300 in frequency) and used the rest of the hypothesized words in our transcripts. We added appropriate lexical entries for these words and built voices for English and Dari.

Table 3 shows the result of syllable and word induction. We see that both for English and Dari, grouping phonemes into syllables decreases the MCD of the new voice. Surprisingly, this difference is very large in the case of Dari, even though the syllable rules were written for English. The voice built for English using CRF word induction has a slightly lower MCD than the syllable method. However, this method does not seem to make much of a difference in the case of Dari. This could be because we used a German word model, and German word rules are quite different from Dari.

Language	Best iteration	Syllables	CRF
English	5.328	5.26	5.25
Dari	4.787	4.165	4.76

Table 3: MCD comparison of voices with syllable and word induction

7. Inducing higher level phonological units

In Section 6, we have demonstrated how syllables and words can be induced from the raw phonetic transcriptions. Here we present an approach that uses the acoustic information, to derive higher order phonological units. While the approach for deriving syllables from phone strings is fairly straight forward across languages (grouping phones together, with the constraint of having one vowel per syllable), the derivation of words from phones is not one-to-one and there is little generalization that is language independent. There is additional complexities for languages with no formal notions of word, or those that are morphologically agglutinative. Here we propose derivation of a more reliable and generalizable phonological unit, the accent group.

We use the broad definition of accent group as being a group of syllables that bears only one intonational accent (a.k.a pitch accent) on them. This definition, while appealing to the idea of metrical feet, does not use pre-defined rules on which syllables should be grouped together, instead opting for a completely data-driven parsing approach (the complete description and training strategy are provided in [24]) as summarized below.

The idea is to analyze the pitch contour in tandem with the underlying syllable sequence and approximate it with a synthetic contour described as a sequence of TILT shapes [23] over parses of syllable groups. The optimal parse on the syllables is one that minimizes the reconstruction error of the target pitch contour. A stochastic context free grammar is trained on such parses of accent groups, so as to allow prediction of accent groups for unseen sequences of syllables. In order to uniquely identify syllables, we tag each syllable with the vowel name, the onset and coda categories as described in [24] (e.g: *syl_onsettype_codatype_vowel*). These categories are only a few in number and yet are language independent, allowing us to use this approach for arbitrary new languages here. This is illustrated in Figure 5 below.

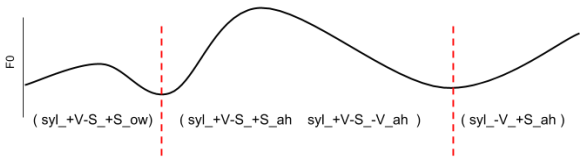


Figure 5: Acoustically derived parse over syllables into accent groups

Given such parses derived acoustically from the pitch contours on all of training data, a grammar is trained to predict parses of unseen sequences of tagged syllables. This is further improved with decision trees about the positional information of each syllable, so as to reliably estimate for each syllable boundary, if there is a accent group boundary, or not.

8. Subjective evaluation for intelligibility

From our objective results mentioned earlier, we saw that the voices built using syllables were better than the voice built on the best iteration using phonemes. Word induction seemed to help in the case of English, but not Dari.

To test this subjectively, we conducted A/B listening tests comparing the voice having the lowest MCD and the voice with syllable units for both English and Dari to see if grouping phonemes together into syllables was perceptually better. Table 5 lists the results from tests on English and Dari. In both cases, we see that participants preferred the voice with syllabified transcripts significantly more than the best iteration.

Language	# participants	Best iteration	Syllable	Can't say
English	7	4%	68%	28%
Dari	5	6%	72%	22%

Table 5: Results of listening tests for English and Dari

In order to test the higher level phonological units i.e. the words and Accent Groups, we built synthetic voices as described in the previous sections. Unseen sequences of sentences in syllabified English were synthesized by each of the above methods i) Syllable transcripts, ii) CRF induced words and iii) Automatically detected Accent Groups. In addition, we also compared the system with Accent Groups to the best iteration from our baseline approach which has no induction of higher level units. 10 sentences of each system were compared in pairs by 10 English listeners who were asked to make a preference to one of the two stimuli. The results are shown in the Table 6.

Voice A	Voice B	Prefer A	Prefer B	Can't say
Best iteration	Accent Group	12%	78%	10%
Induced words (CRF)	Accent Group	22%	70%	8%
Syllable	Accent Group	47%	43%	10%

Table 6: Results of listening tests for English

The results indicate that significant gains can be obtained by induction of the speech-derived accent group units, as opposed to word derivations through CRFs over phoneme transcriptions. While it is encouraging that the Accent Group voices perform comparably, syllable voices remain the most reliable units that can be induced in the current setting. This is perhaps due to the unavailability of sufficient data, or features that effectively capture the contextual information in building voices using higher levels of phonology.

9. Conclusion and future work

In this paper, we applied our iterative cross-lingual decoding technique to eight languages from various language families. We saw that with as little as half an hour of speech, we could get improvements in MCD over the baseline decoded transcripts.

We also used heuristics to syllabify the phoneme transcripts and a CRF to automatically induce word like units, which led to higher quality voices, both objectively and subjectively. In addition, we described a method to use acoustic information to identify accent groups to create higher level phonological units which may help improve the quality of synthesis. Our results indicate that inducing such units leads to a large improvement in both MCD and subjective preference.

From our initial experiments on building SMT systems from the source language to the target learned transcript, knowing where the word boundaries are can prove to be critical for good translation. We plan to explore other methods to automatically derive higher level units from text and acoustics.

In the future, we also plan to explore using combinations of multiple acoustic and language models instead of relying on a single model for the initial decoding pass. We also realize the importance of the initial phoneset and plan to explore more principled methods of pruning phonesets at each iteration. The next stage of this work is to extend it to use machine translation to provide a usable writing system for languages without a standardized orthography.

Acknowledgment

This research was supported in part by a Google Research Award "Text-to-Speech in New Languages without the Text"

References

[1] Bowen Zhou, Laurent Besacier, and Yuqing Gao, "On Efficient Coupling of ASR and SMT for Speech Translation," in Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing, Honolulu, HI, USA, April 2007, vol. 4, pp. 101–104.

[2] Nicola Bertoldi, Richard Zens, and Marcello Federico, "Speech Translation by Confusion Network Decoding," in Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing, Honolulu, HI, USA, April 2007, vol. 4, pp. 1297–1300.

[3] Pablo Daniel Agüero, Jordi Adell, and Antonio Bonafonte, "Prosody Generation for Speech-to-Speech Translation," in Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing, Toulouse, France, May 2006.

[4] Vivek Kumar Rangarajan Sridhar, Srinivas Bangalore, and Shrikanth S. Narayanan, "Factored Translation Models for enriching Spoken Language Translation with Prosody," in Proceedings of Interspeech, Brisbane, Australia, September 2008, pp. 2723–2726.

[5] Laurent Besacier, Bowen Zhou, and Yuqing Gao, "Towards Speech Translation of non Written Languages," in Proceedings of the IEEE Workshop on Spoken Language Technology, Palm Beach, Aruba, December 2006, pp. 222–225.

[6] Sebastian Stüker and Alex Waibel, "Towards Human Translations Guided Language Discovery for ASR Systems," in Proceedings of Spoken Language Technologies for UnderResourced Languages, 2008.

[7] Zeeshan Ahmed, Jie Jiang, Julie Carson-Berndsen, Peter Cahill, and Andy Way, "Hierarchical Phrase-Based MT for Phonetic Representation-Based Speech Translation," in Proceedings of the tenth Biennial Conference of the Association for Machine Translation in the Americas (AMTA), San Diego, CA, October 2012.

[9] Felix Stahlberg, Tim Schlippe, Stephan Vogel, and Tanja Schultz, "Word Segmentation Through Cross-Lingual Word to-Phoneme

Alignment," in Proceedings of IEEE Workshop on Spoken Language Technology, Miami, FL, December 2012.

[10] George Zavaliagkos and Thomas Colthurst, "Utilizing Untranscribed Training Data to Improve Performance," in Proceedings of The DARPA Broadcast News Transcription and Understanding Workshop, 1998.

[11] Sukhada Palkar, Alan W Black, and Alok Parlikar, "Text-to Speech for Languages without an Orthography," in Proceedings of the 24th International conference on Computational Linguistics, Mumbai, India, December 2012.

[12] Sunayana Sitaram, Sukhada Palkar, Alok Parlikar, and Alan W Black, "Bootstrapping Text-to-Speech for Speech Processing in Languages without an Orthography" in Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP), Vancouver, Canada, May 2013.

[13] Micha Elsner, Sharon Goldwater, and Jacob Eisenstein, "Bootstrapping a Unified Model of Lexical and Phonetic Acquisition," in Proceedings of Association for Computational Linguistics, Jeju island, Korea, July 2012.

[14] Alan W Black and Keiichi Tokuda, "Blizzard Challenge -- 2005: Evaluating corpus-based speech synthesis on common datasets" Interspeech, Lisbon, Portugal, 2005.

[15] T. Schultz, A. W. Black, S. Badaskar, M. Hornyak, and J. Kominek, "Spice: Web-based tools for rapid language adaptation in speech," in Proceedings of INTERSPEECH, Antwerp, Belgium, August 2007.

[16] Paul Placeway, Stanley F. Chen, Maxine Eskenazi, Uday Jain, Vipul Parikh, Bhiksha Raj, Ravishankhar Mosur, Roni Rosenfeld, Kristie Seymore, Matthew A. Siegler, Richard M. Stern, and Eric Thayer, "The 1996 Hub-4 Sphinx-3 System," in Proceedings of the DARPA Speech Recognition Workshop, 1996

[17] Alan W Black and Paul Taylor, "The Festival Speech Synthesis System: system documentation," Tech. Rep., Human Communication Research Centre, University of Edinburgh, January 1997.

[18] Alan W Black, "CLUSTERGEN: A Statistical Parametric Synthesizer using Trajectory Modeling," in Proceedings of Interspeech, Pittsburgh, Pennsylvania, September 2006, pp. 194–197.

[19] Philipp Koehn, "Europarl: A Parallel Corpus for Statistical Machine Translation," in Proceedings of Machine Translation Summit, Phuket, Thailand, September 2005, pp. 79–86.

[20] Alok Parlikar, "TestVox: Web-based Framework for Subjective Evaluation of Speech Synthesis," Opensource Software, 2012.

[21] Mikiko Mashimo, Tomoki Toda, Kiyohiro Shikano, and Wilhelm Nicholas Campbell, "Evaluation of Cross-Language Voice Conversion Based on GMM and Straight," in Proceedings of Eurospeech, Aalborg, Denmark, September 2001, pp. 361–364.

[22] Taku Kudoh, "Crf++", Software, http://crfpp.sourceforge.net/, 2007.

[23] P Taylor, "Analysis and synthesis of intonation using the tiltmodel," Journal of the Acoustical Society of America, vol. 1073, pp. 1697–1714, 2000.

[24] Gopala Krishna Anumanchipalli, Luis C Oliveira, Alan W Black, " Accent Group Modeling for Improved Prosody in Statistical Parametric Speech Synthesis", in proceedings of IEEE ICASSP 2013, Vancouver, Canada, 2013.

Unsupervised and lightly-supervised learning for rapid construction of TTS systems in multiple languages from 'found' data: evaluation and analysis

O. Watts[1], *A. Stan*[2], *R. Clark*[1], *Y. Mamiya*[1], *M. Giurgiu*[2], *J. Yamagishi*[13], *S. King*[1]

[1]The Centre for Speech Technology Research, University of Edinburgh, UK
[2]Communications Department, Technical University of Cluj-Napoca, Romania
[3]National Institute of Informatics, Japan

{adriana.stan, mircea.giurgiu}@com.utcluj.ro, Simon.King@ed.ac.uk,
{owatts, Yoshitaka.Mamiya, robert, jyamagis}@inf.ed.ac.uk

Abstract

This paper presents techniques for building text-to-speech front-ends in a way that avoids the need for language-specific expert knowledge, but instead relies on universal resources (such as the Unicode character database) and unsupervised learning from unannotated data to ease system development. The acquisition of expert language-specific knowledge and expert annotated data is a major bottleneck in the development of corpus-based TTS systems in new languages. The methods presented here side-step the need for such resources as pronunciation lexicons, phonetic feature sets, part of speech tagged data, etc. The paper explains how the techniques introduced are applied to the 14 languages of a corpus of 'found' audiobook data. Results of an evaluation of the intelligibility of the systems resulting from applying these novel techniques to this data are presented.
Index Terms: multilingual speech synthesis, unsupervised learning, vector space model, text-to-speech, audiobook data

1. Introduction

Collecting and annotating the data necessary for training a corpus-based text-to-speech (TTS) conversion system in a new language requires considerable time and expert knowledge. Conventionally, audio data for training a synthesiser *back-end* (or waveform generator) will be gathered during a specially-arranged recording session. For this, a recording script must be prepared, a suitable studio must be found, a voice talent must be recruited and speech recording must be carefully supervised. One of the primary goals of the *Simple4All*[1] project is to reduce the time and expert knowledge needed to produce new TTS systems. In [1] we presented a toolkit – developed as part of this project – for segmenting and aligning existing freely-available recordings (audiobooks), circumventing to some extent the need to engineer purpose-recorded speech corpora. The outcome of applying those tools to audiobooks in 14 languages is what we have released under the name of the *Tundra corpus*.

However, the problems associated with TTS data-collection do not stop when we have obtained transcribed speech data for training a synthesiser back-end. TTS systems also require a *front-end* (or text analysis module), which accepts input text and outputs a representation of an utterance suitable for input into the back-end. TTS systems generally represent utterances in terms of units and features based on linguistic knowledge, such as phonemes, syllables, lexical stress, phrase boundaries etc. The components of the front-end that predict these from input text are either made up of hand-written rules or statistical modules; acquiring the expert knowledge required either to manually specify those rules, or to annotate a learning sample on which to train the statistical models, represents a major obstacle to creating a TTS system for a new target language and requires highly specialised knowledge. Such non-trivial tasks include, for example, specifying a phoneme-set or part of speech (POS) tag-set for a language where one has not already been defined; annotating plain text with POS tags, as required to train a POS tagger and annotating the surface forms of words with phonemes to build a pronunciation lexicon.

The toolkit we are developing in *Simple4All* includes tools for constructing TTS front-ends which make as few implicit assumptions about the target language as possible, and which can be configured with minimal effort and expert knowledge to suit arbitrary new target languages. To this end, the modules rely on resources which are intended to be universal, such as the Unicode character database, and employ unsupervised learning so that unlabelled text resources can be exploited without the need for costly annotation. The current paper presents these tools and explains how they were applied to the data of the Tundra corpus to produce TTS systems in 14 languages. We present the results of a listening test of the intelligibility of those systems, and thus evaluate the entire pipeline implemented by our toolkit, which begins with raw found data and ends with trained TTS systems. An initial public version of tools for this whole pipeline (for segmenting and aligning found data and for producing TTS systems with minimal expert knowledge) is due to be released in November 2013.

In prior work addressing the bottleneck in TTS system construction represented by the front-end, unified systems aimed at producing complete systems have generally taken the strategy of providing infrastructure to ease the collection by non-experts of the conventional resources necessary for system construction. This infrastructure might take the form of user-friendly development environments [2], or training and on-going support [3]. Prior work has also presented unsupervised methods for building systems based on letters rather than phonemes [4, 5], induction of phone-sets [6, 7], syllable-like units [8, 9], or lexicons [10]. However, this work has not been presented as an integrated framework for producing end-to-end TTS systems. Furthermore, despite the significant work on unsupervised learning in Natural Language Processing [11, 12] and Information Retrieval [13, 14], potentially useful techniques developed in those fields have not been applied to the problem of TTS front-end induction.

[1] www.simple4all.org/

2. Database

The Tundra corpus [1] is a standardised multilingual corpus designed for text-to-speech research with imperfect or found data. It consists of 14 audiobooks in 14 different languages (Bulgarian, Danish, Dutch, English, Finnish, French, German, Hungarian, Italian, Polish, Portuguese, Romanian, Russian and Spanish) and amounts to approximately 60 hours of speech. A complete list of the audiobooks with their sources and durations can be found here `http://tundra.simple4all.org`.

The corpus provides utterance-level alignments obtained with a lightly supervised process described in [15] and [16]. The accuracy of the alignment method, as described in [16] is of 7% SER and 0.8% WER, therefore some light post-processing is required in order to eliminate some of the erroneous utterances. Initial segmentation of the audiobooks into utterance-size chunks was performed using the lightly supervised GMM-based VAD described in [17]. As most of the used audiobooks are recorded in non-specialised environments, the speech data underwent a light cleaning process: normalising the DC offset, applying a multi-band noise gate removal and an RMS-based deverberation method, as described in [1].

3. System Construction

For each of the 14 languages of the Tundra corpus, a TTS system was trained with no reliance of language-specific expertise. Although speaker and recording differences mean that meaningful comparison between languages is difficult, we wished to make the training conditions for the 14 voices as uniform as possible. Therefore, we selected a 1 hour subset of each of the languages' data on which to train voices for this evaluation: the method of data selection we used is explained in Section 3.1. Then text analysis and waveform generation components were trained on that selected data as explained in Sections 3.2 and 3.3, respectively.

3.1. Lightly-supervised data selection

Our principal current interest in audiobook data is that it presents a source of 'found' data from which TTS training databases can be harvested without the need to construct a recording script, recruit a native speaker of the target language, and supervise the recording of a script from scratch. In the present work, therefore, we ignore the other possible advantage of using audiobook data: that harnessing the variety of speaking styles present in audiobooks might enable us to produce less 'mechanical'-sounding TTS systems. Although this is a longer-term goal, we here follow an approach similar to the one presented in [18], which aims to select a neutral subset of a database containing *diverse* speech. In that paper, 9 utterance-level acoustic features are used along with several textual cues to exclude diverse speech from the training set. Thresholds over these features are set manually by the system builder to exclude non-neutral utterances.

For the current work we perform utterance selection using an active learning approach, with uncertainty sampling [19]. Rather than being required to tune thresholds manually, the system builder is presented with example utterances and asked to indicate whether or not they are spoken in a neutral style. The interface therefore insulates the user from the details of the features used, and lets the user focus on what should be key: their intuitive response to hearing speech samples. The procedure we used is as follows:

1) **Feature extraction** First, frame-level features (F_0, energy and spectral tilt – approximated by 1st mel cepstral coefficient) are obtained, from which utterance-level features are computed. The fact that no thresholds need to be manually tuned means that we can afford to use a great many more features than the 9 employed in [18]. Our feature set is based on the one described in [20]: we compute mean, standard deviation, range, slope, minimum and maximum (6-level factor) for $F0$, spectral tilt, and energy (3-level factor) in the following sub-segments of each utterance: entire utterance, 1st and 2nd halfs, all 4 quarters, first and last 100ms, first and last 200ms (11-level factor), giving a total of 198 features.

2) **Initial labelling** The user is presented with the audio of s randomly-selected *seed utterances* from the whole corpus (via a text-based user interface) and asked to label them *keep* or *discard* – utterances are labelled with the user's decision.

3) **Classifier training** A classifier is trained on the labelled examples. Our choice of classifier is a bagged ensemble of decision trees [21] because it can be trained quickly (allowing online active learning in real time), is robust against noisy features and able to accept unnormalised input variables, and mixtures of discrete and continuous input variables (allowing a great many different acoustic features to be used, and different types of features), allows the space of utterances to be partitioned recursively (enabling complex interactions between features to be detected), and provides robust estimates of class probabilities (important for step 4).

4) **Uncertainty sampling** The set of u uncertain examples (utterances about which the classifier is most uncertain – in the present case, the utterances which have closest to 0.5 *keep* probability). The utterances in this set are presented to the user for labelling.

5) Steps 3 and 4 are repeated as many times as time allows.

6) The set of utterances either labelled *keep* by the user are kept for training, as well as enough of the utterances to which the trained classifier gives the highest *keep* probability to, to make up the desired quantity of training data.

For the work presented here, s was set to 15 and u was set to 1. That is, the user was asked to provide 15 labels at the outset, and presented with a single uncertain example at each iteration. The stopping criterion we used in this work was to limit the number of iterations to 15 – in the present, utterance selection time was limited to approximately 20 minutes per language, and 15 was found to be a reasonable number of iterations in that time. Informal comparison suggested the approach outlined is beneficial for this task, but in ongoing work we are testing this rigorously and comparing uncertainty sampling with random sampling, as well as applying our active learning tool to other TTS tasks.

3.2. Front-end construction with unsupervised learning

The TTS front-end building tools used for this work are based on ideas outlined in [22] and applied to Spanish TTS in [23]. Input to the system consists of the audio of utterances selected as described in Section 3.1, together with their text transcription (aligned at the utterance level): in the present case, these are taken from the Tundra corpus, and had been obtained as summarised in Section 2. As an additional input, 5 million words of running text data were obtained from Wikipedia in the target languages for construction of the word- and letter-representations described below.

Text which is input to the system is assumed to be UTF-8 encoded: given UTF-8 text, text processing is fully automatic and makes use of a theoretically universal resource: the Uni-

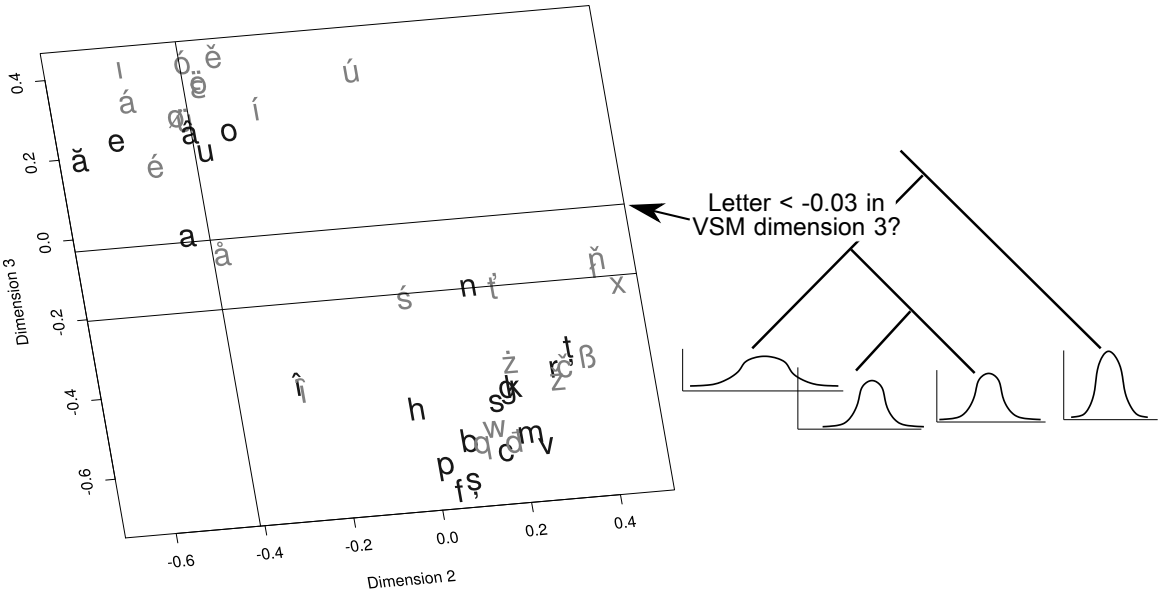


Figure 1: Use of a letter space to replace phonetic knowledge in decision-tree based state-tying. Shown here are 2 dimensions of the actual letter space induced in training the Romanian system described in the paper. The 3 lines bisecting the space represent the 3 questions actually asked in the uppermost fragment (first three 'generations') of the state-tying decision tree for the central state of the model for spectral envelope features. Letters shown in black are 'heard' by the system (i.e. are present in the transcriptions of the audio training data) but ones shown in grey are only 'seen' (i.e. appear only in textual training data) and are mainly foreign language letters.

code database. Unicode character properties are used to tokenise the text and characterise tokens as words, whitespace, punctuation etc. Our modules have so far been successfully applied to a variety of alphabetic (Latin-based, Cyrillic) and alphasyllabic (Brahmic) scripts. Our front-ends currently expect text without abbreviations, numerals, and symbols (e.g. for currency) which require expansion; however, the lightly supervised learning of modules to expand such non-standard words is an active topic of research [24], and we hope to integrate such modules into our toolkit in the near future.

A letter-based approach is used, in which the names of letters are used directly as the names of speech modelling units (in place of the phonemes of a conventional front-end). This has given good results for languages with transparent alphabetic orthographies such as Romanian, Spanish and Finnish, and can give acceptable results even for languages with less transparent orthographies, such as English [22, 4, 5, 7].

The induced front-ends make use of no expert-specified categories of letter and word, such as phonetic categories (vowel, nasal, approximant, etc.) and part of speech categories (noun, verb, adjective, etc.). Instead, features that are designed to stand in for such expert knowledge but which are derived fully automatically from the distributional analysis of unannotated text (speech transcriptions and Wikipedia text) are used. The distributional analysis is conducted via vector space models (VSMs); the VSM was originally applied to the characterisation of documents for purposes of Information Retrieval. VSMs are applied to TTS in [22], where models are built at various levels of analysis (letter, word and utterance) from large bodies of unlabelled text. To build these models, co-occurrence statistics are gathered in matrix form to produce high-dimensional representations of the distributional behaviour of e.g. word and letter types in the corpus. Lower-dimensional representations are obtained by approximately factorising the matrix of raw co-occurrence counts by the application of slim singular value decomposition. This distributional analysis places textual objects in a continuous-valued space, which is then partitioned by decision tree questions during the training of TTS system components such as acoustic models for synthesis or decision trees for pause prediction. For the present voices, a VSM of letters was constructed by producing a matrix of counts of immediate left and right co-occurrences of each letter type, and from this matrix a 5-dimensional space was produced to characterise letters. Token co-occurrence was counted with the nearest left and right neighbour tokens (excluding whitespace tokens); co-occurrence was counted with the most frequent 250 tokens in the corpus. A 10-dimensional space was produced to characterise tokens.

Two dimensions of the letter space induced in training the Romanian system are shown in Figure 1. It can be seen that in these dimensions of the space, vowel and consonant symbols are clearly separable. When a decision tree for clustering acoustic model states is built and allowed to query items' positions in these 2 dimensions, it can use all partitions of the space orthogonal to its axes. A decision tree question such as *Is the letter's value in VSM dimension 3 < -0.03?* is very nearly equivalent to a question based on linguistic knowledge such as *Is the letter a consonant?* The categories of vowel and consonant are useful for clustering acoustic models, and so decision trees actually built using this space use such partitions of the space: the 3 lines shown bisecting the space in the figure represent the 3 questions actually asked in the uppermost fragment (first three 'generations') of the state-tying decision tree for the central state of the model for spectral envelope features.

Distributional analysis places linguistic or textual units in a continuous space which is then partitioned on acoustic evidence. The space constrains the possible groupings of objects that can be considered during decision tree growing. Distributional analysis also allows splits made to generalise to items

that are 'seen' by the system in text data but not 'heard' in the audio data. This is most obviously useful where units such as words are concerned, where many items not present in the training speech corpus are likely to occur at run-time. It can, however, also be useful where letters are concerned, and some examples that illustrate our models' ability to generalise beyond what is heard can be seen in the letter space shown in Figure 1. There, letters shown in black are 'heard' by the system but ones shown in grey are only 'seen' – these are mainly due to foreign language words within Romanian Wikipedia entries. It can be seen that unheard foreign vowels such as *á* and *ö* are suitably placed near the Romanian vowels, and unheard consonants such as *ß* and *q* are placed near the consonants that are actually heard. Splits such as those shown – made only on the basis of the heard items – therefore generalise to unheard items. In the case of letters, this allows rare and foreign letters to be handled despite their absence in the transcriptions of acoustic training data. It can also allow better handling of non-standard spellings: in the case of the vowel *î* (*i* with circumflex), there is a variant (with inverted breve instead of circumflex) which is not present in any of the speech transcriptions but which is used in a few Wikipedia articles. From Figure 1 it can be seen that almost identical representations are learned for these two letters, meaning a decision tree built using those representations will be able to handle the variant form correctly at run-time, even though no instances of that variant were seen in the transcription of the speech training corpus.

The front ends make use of decision trees to predict pauses at the junctures between words. Data for training these trees are acquired automatically by force-aligning the training data with their transcriptions, and allowing the optional insertion of silence between words. The independent variables used by the trees are whether words are separated by punctuation or space, and the VSM features of the tokens preceding and following the juncture.

A rich set of contexts is created using the results of the analysis described here for each letter token in the database. Features include the identity of the letter and the identities of its neighbours (within a 5-letter window), the VSM values of each of those letters, and the distance from and until a word boundary, pause, and utterance boundary. In the current systems, word VSM features are not included directly in the letter contexts, but are used by the decision tree for predicting pauses at runtime.

3.3. Back-end construction

For training the waveform generation modules for the 14 voices, the waveforms of the training corpora were parameterised almost as described in [25]. The one difference is that instead of the committee of different pitch-trackers used in the earlier work, pitch tracks obtained from a glottal source signal estimated by glottal inverse filtering [26] were used for their greater accuracy.

For all systems, speaker-dependent acoustic models were built from this parameterised speech data and the annotation described in Section 3.2, using the speaker-dependent model-building recipe described in [27].

Static and interactive demos of the resulting voices are available at `http://tundra.simple4all.org/demo`. A screen shot of the geographically-organised demo page is shown in Figure 2.

Figure 2: Demo screenshot: this geographical interface to voices can be found at `http://tundra.simple4all.org/demo`.

4. System Evaluation

4.1. Procedure

We are primarily interested in having our systems produce *intelligible* speech; evaluation therefore focused on the intelligibility of TTS output as measured by the word and letter error rates of listeners' transcriptions of those outputs. Conventionally in TTS evaluation, listeners are asked to transcribe semantically unpredictable sentences (SUS) [28]. However, such SUS are not currently available in all the Tundra languages and it is not trivial to construct new SUS, and so we resorted to using short natural sentences from the held-out test sets of the Tundra corpus.

For all 14 Tundra languages, 40 sentences were manually segmented from the held-out chapters of the relevant audiobook. Note that these test sets are distributed with the Tundra corpus, and so the results presented below can be considered benchmarks for future work. An attempt was made to select sentences of 6–8 words in order to make the inherent difficulty of transcription as uniform as possible. However, in some languages these thresholds had to be relaxed; Table 1 gives statistics of test-sentence lengths in all languages.

Subjects for the evaluation were recruited through a web-based crowdsourcing service. The advert for the evaluation specified that native speakers of the relevant language were required; in addition, participation in each part of the evaluation was restricted to users registered in countries where the relevant language is an official or majority language. We attempted to recruit listeners to evaluate all 14 systems built. However, as the option to restrict participation to workers registered in Denmark, Finland and Hungary was not available in the service we used, listening test for only 11 of the systems were publicised. The number of responses from participants varied greatly between languages. At the time of writing, responses from a sufficient number of listeners (25+) had been collected in only 5 of the languages (Bulgarian, English, Italian, Polish and Romanian) . Results for these five languages are presented here; evaluation of the remaining voices is left for future work.

In all languages, two conditions were evaluated: the natural speech of the natural sentences from the test set, and the

Table 1: Statistics of Tundra test-sentence lengths (number of words)

Language	Mean	Standard deviation
German	6.63	0.87
Finnish	6.8	0.91
Bulgarian	6.85	0.83
English	6.88	0.94
Italian	6.9	0.87
Polish	6.95	0.88
Hungarian	7.05	0.81
Russian	7.13	1.18
Danish	7.4	1.19
Portuguese	8.08	1.47
Dutch	8.1	2.15
Romanian	8.55	1.97
French	8.58	1.96
Spanish	8.8	1.65

TTS system reading the same text. In the four languages of the Simple[4]All consortium members (including two of the languages for which results are presented here: Romanian and English), however, SUS were available, and so for those languages a third condition was evaluated: the TTS system producing SUS texts. This is designed to provide a way of broadly gauging the relative difficulty of transcribing natural and SUS sentences, although language and text differences mean it is obviously not advisable to treat extrapolation of the differences to the remaining languages with any great confidence.

The evaluation was run as a set of webpages where participants were asked – using headphones – to listen to the samples and to type in what they heard. Multiple listens were allowed as some of the the natural sentences were longer than the short SUS we would typically use. For the first two conditions, a balanced design was used so that each listener heard each utterance text only once, while each text was heard an equal number of times in both conditions over the whole evaluation. Each listener heard 20 sentences spoken in each condition. For English and Romanian where the SUS condition was also included, listeners heard a further set of 20 SUS sentences.

4.2. Results

Word error rates for the first 2 conditions are shown in Figure 3. For all languages besides English, a similar pattern can be observed: listeners' transcriptions of natural speech attain a WER of 8–12%, and in all cases the TTS system attain WERs approximately 1.5 times worse. This is consistent with the difference between WERs for natural speech and decent benchmark systems in larger scale evaluations on standard corpora. For example, natural speech and the Festival benchmark system attained WERs of 17% and 25% respectively in the 2011 Blizzard Challenge evaluation [29]. The results for English are the exception to the general pattern: the WER for synthetic speech is over 4 times worse than that of natural speech. From prior knowledge and from looking at listeners' transcriptions, it seems clear that this is due to the fact that TTS is based on letters in a language with such an opaque letter-to-sound relationship. In all languages except Polish, the difference between the first two conditions (natural speech and TTS) found to be statistically significant (with $\alpha = 0.05$) using the bootstrap procedure of [30].

As expected, WERs for the SUS sentences are much higher than those for natural sentences: 24.8% and 69.4% for Romanian and English, respectively.

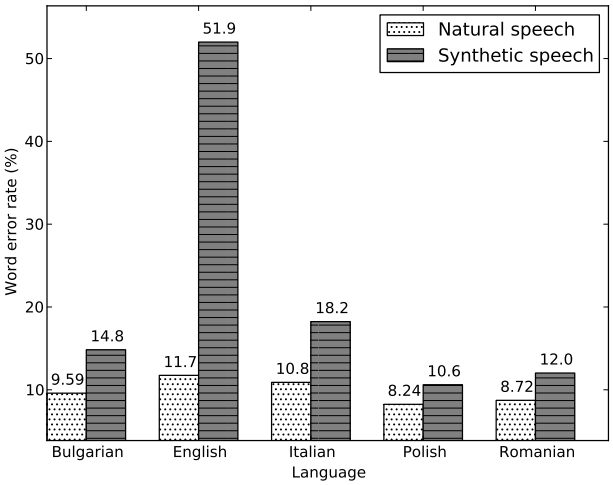


Figure 3: Word error rates for TTS systems and natural speech for 5 of the 14 systems built from the Tundra corpus.

5. Conclusions

We have presented tools for building TTS front-ends in a way that exploits unsupervised learning techniques to side-step the need for language-specific expert knowledge and resources such as pronunciation lexicons, phoneme inventories and part of speech taggers. We have shown how the tools were applied to the languages of the Tundra corpus to produce TTS systems in 14 languages. As we had previously built the Tundra corpus from found data using minimal supervision and language specific knowledge, these TTS systems represent the output of our entire pipeline of tools, and show the type of voice which any interested developer should be able to build using our toolkit (which will be made freely available) despite a lack of language-specific or speech technology expertise, if a source of speech and text data can be found. Five of the voices were evaluated in a listening test for intelligibility, which we consider to show that systems of reasonable quality can be built by applying our tools to publicly available audiobook data, assuming orthographies of similar transparency to those of Bulgarian, Italian, Polish and Romanian. While evaluation of the remaining systems that can be heard in the demo is still ongoing, the results for five languages published here – having been obtained from a standardised, publicly available corpus – are intended to be useful benchmarks against which future work can be compared.

6. Acknowledgements

The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement Nº 287678.

The research presented here has made use of the resources provided by the Edinburgh Compute and Data Facility (ECDF: `http://www.ecdf.ed.ac.uk`). The ECDF is partially supported by the eDIKT initiative (`http://www.edikt.org.uk`).

Thanks to Vasilis Karaiskos for setting up the webpages for the listening test.

7. References

[1] A. Stan, O. Watts, Y. Mamiya, M. Giurgiu, R. A. J. Clark, J. Yamagishi, and S. King, "TUNDRA: A Multilingual Corpus of Found Data for TTS Research Created with Light Supervision," in *Proc. of Interspeech (accepted)*, 2013.

[2] J. Kominek, T. Schultz, and A. W. Black, "Voice building from insufficient data – classroom experiences with web-based language development tools," in *Proc. 6th ISCA Speech Synthesis Workshop*, Bonn, Germany, 2007, pp. 322–327.

[3] R. Tucker and K. Shalonova, "Supporting the creation of TTS for local language voice information systems," in *Proc. Interspeech 2005*, Lisbon, Portugal, Sep. 2005, pp. 453–456.

[4] A. Black and A. Font Llitjos, "Unit selection without a phoneme set," in *IEEE TTS Workshop 2002*, 2002.

[5] G. Anumanchipalli, K. Prahallad, and A. Black, "Significance of early tagged contextual graphemes in grapheme based speech synthesis and recognition systems," in *Acoustics, Speech and Signal Processing, 2008. ICASSP 2008. IEEE International Conference on*, 31 2008-April 4 2008, pp. 4645–4648.

[6] J. Černocký, "Speech processing using automatically derived segmental units: Applications to very low rate coding and speaker verification," Ph.D. dissertation, Universite Paris-Sud, Dec 1998.

[7] M. P. Aylett, S. King, and J. Yamagishi, "Speech synthesis without a phone inventory," in *Interspeech*, 2009, pp. 2087–2090.

[8] P. Mermelstein, "Automatic segmentation of speech into syllabic units," *The Journal of the Acoustical Society of America*, vol. 58, no. 4, pp. 880–883, 1975.

[9] Z. Xie and P. Niyogi, "Robust acoustic-based syllable detection," in *Proceedings of the ICSLP, International Conference on Spoken Language Processing*, 2006.

[10] J. Kominek, "Tts from zero: Building synthetic voices for new languages," Ph.D. dissertation, Carnegie Mellon University, 2009.

[11] M. Creutz and K. Lagus, "Unsupervised models for morpheme segmentation and morphology learning," *ACM Trans. Speech Lang. Process.*, vol. 4, no. 1, pp. 3:1–3:34, Feb. 2007.

[12] C. Christodoulopoulos, S. Goldwater, and M. Steedman, "Two decades of unsupervised POS induction: How far have we come?" in *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, October 2010, pp. 575–584.

[13] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet allocation," *J. Mach. Learn. Res.*, vol. 3, pp. 993–1022, March 2003.

[14] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman, "Indexing by latent semantic analysis," *Journal of the American Society for Information Science*, vol. 41, pp. 391–407, 1990.

[15] A. Stan, P. Bell, and S. King, "A grapheme-based method for automatic alignment of speech and text data," in *Proc. IEEE Workshop on Spoken Language Technology, Miami, Florida, USA*, 2012.

[16] A. Stan, P. Bell, J. Yamagishi, and S. King, "Lightly Supervised Discriminative Training of Grapheme Models for Improved Sentence-level Alignment of Speech and Text Data," in *Proc. of Interspeech (accepted)*, 2013.

[17] Y. Mamiya, J. Yamagishi, O. Watts, R. A. Clark, S. King, and A. Stan, "Lightly Supervised GMM VAD to use Audiobook for Speech Synthesiser," in *Proc. ICASSP*, 2013.

[18] N. Braunschweiler and S. Buchholz, "Automatic sentence selection from speech corpora including diverse speech for improved HMM-TTS synthesis quality," in *Proc. Interspeech*, Florence, Italy, Aug. 2011, pp. 1821–1824.

[19] D. D. Lewis and W. A. Gale, "A sequential algorithm for training text classifiers," in *Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval*.

[20] G. Murray, S. Renals, and M. Taboada, "Prosodic correlates of rhetorical relations," in *Proceedings of HLT/NAACL ACTS Workshop, 2006, New York City, USA*, Jun. 2006.

[21] L. Breiman, "Bagging predictors," *Machine Learning*, vol. 24, pp. 123–140, 1996.

[22] O. Watts, "Unsupervised learning for text-to-speech synthesis," Ph.D. dissertation, University of Edinburgh, 2012.

[23] J. Lorenzo-Trueba, O. Watts, R. Barra-Chicote, J. Yamagishi, S. King, and J. M. Montero, "Simple4All proposals for the Albayzin Evaluations in Speech Synthesis," in *Proc. Iberspeech 2012*, 2012.

[24] R. San-Segundo, J. M. Montero, V. Lopez-Ludeña, and S. King, "Detecting acronyms from capital letter sequences in Spanish," in *Proc. Interspeech*, Portland, Oregon, USA, Sep. 2012.

[25] J. Yamagishi and O. Watts, "The CSTR/EMIME HTS System for Blizzard Challenge," in *Proc. Blizzard Challenge 2010*, Sep. 2010.

[26] T. Raitio, A. Suni, J. Yamagishi, H. Pulakka, J. Nurminen, M. Vainio, and P. Alku, "HMM-based speech synthesis utilizing glottal inverse filtering," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 19, no. 1, pp. 153–165, Jan. 2011.

[27] H. Zen, T. Toda, M. Nakamura, and K. Tokuda, "Details of Nitech HMM-based speech synthesis system for the Blizzard Challenge 2005," *IEICE Trans. Inf. & Syst.*, vol. E90-D, no. 1, pp. 325–333, Jan. 2007.

[28] C. Benoit, M. Grice, and V. Hazan, "The SUS test: A method for the assessment of text-to-speech synthesis intelligibility using Semantically Unpredictable Sentences," *Speech Communication*, vol. 18, no. 4, pp. 381 – 392, 1996.

[29] S. King and V. Karaiskos, "The blizzard challenge 2011," in *Proc. Blizzard Challenge 2011*, sep 2011.

[30] M. Bisani and H. Ney, "Bootstrap estimates for confidence intervals in ASR performance evaluation," in *Proc. ICASSP '04*, vol. 1, 2004, pp. 409–12.

A phonetic-contrast motivated adaptation to control the degree-of-articulation on Italian HMM-based synthetic voices

Mauro Nicolao[1], *Fabio Tesser*[2], *Roger K. Moore*[1]

[1]Speech and Hearing Group, Dept. Computer Science, University of Sheffield, UK
[2]Institute of Cognitive Sciences and Technologies, National Research Council, Padova, Italy
m.nicolao@dcs.shef.ac.uk, fabio.tesser@pd.istc.cnr.it, r.k.moore@dcs.shef.ac.uk

Abstract

The effectiveness of phonetic-contrast motivated adaptation on HMM-based synthetic voices was previously tested on English successfully. The aim of this paper is to prove that such adaptation can be exported with minor changes to languages having different intrinsic characteristics. The Italian language was chosen because it has no obvious phonemic configuration towards which human speech tend when hypo-articulated such as the mid-central vowel (schwa) for English. Nonetheless, low-contrastive attractors were identified and a linear transformation was trained by contrasting each phone pronunciation with its nearest acoustic neighbour. Different degree of hyper and hypo articulated synthetic speech was then achieved by scaling such adaptation along the dimension identified by each contrastive pair. The Italian synthesiser outcome adapted with both the maximum and the minimum transformation strength was evaluated with two objective assessments: the analysis of some common acoustic correlates and the measurement of a intelligibility-in-noise index. For the latter, signals were mixed with different disturbances at various energy ratios and intelligibility was compared to the standard-TTS generated speech. The experimental results proved such transformation on the Italian voices to be as effective as those on the English one.

Index Terms: hypo/hyper-articulated speech synthesis, Italian HMM-based synthesis, intelligibility enhancement, speech adaptation, statistical parametric speech synthesis.

1. Introduction

The observation that human talkers modify their speech production according to the environmental condition was established more than a century ago by Lombard [1].

According to theories such as the Lindblom's H&H (hypo-hyper) theory of speech production [2], such adjustments are controlled by the need of maximising the effectiveness of the communication minimising the effort at the same time. Mainly, these are driven by constant monitoring their effectiveness.

Moore's PRESENCE [3] model was one of the first attempts to import human-inspired ideas into a computational model, which could be also included in a complex automatic speech communication system. Following this comprehensive model, a first reactive synthesiser that could react to environmental disturbances by enhancing the contrast between competing phones was proposed [4]. This work was mainly motivated by the lack of expressivity observed in standard text-to-speech systems. The few expressive synthetic speech synthesisers are tuned to specific needs and therefore not able to react to different environmental conditions dynamically.

A first complete Computational model for Hyper and Hypo articulated speech synthesis (C2H), which would monitor its output in order to maximise the intelligibility in noise, was proposed in [5] and its effectiveness was tested with an HMM-based English speech synthesiser.

Recently, other approaches, which focus mainly on sound quality modifications to maximise the speech audibility in noise, have tackled the problem from different angles: synthetic Lombard-speech generation by manipulating the glottal source signal [6]; feature optimisation to increase the intelligibility of the parametric generated speech [7]; hyper and hypo-articulated speech synthesis by interpolating between 'ad hoc' recorded corpora [8].

Aim of this paper is to apply the C2H test adaptation technique [5], which relied upon the reduction/expansion of the vowel space using the schwa [ə] vowel, with minor changes to Italian. This language differs from English because it has no such clear low-energy attractors for hypo-articulated speech in its phonetic inventory.

In the following sections, the details of the adaptation on two existing Italian HMM-based voices (a female and a male one) are presented. Particular emphasis is used to describe the training process and to motivate the low-contrastive attractor choice. The Italian synthesiser outcome adapted both with the maximum and the minimum transformation strength was evaluated with two objective analyses: the extraction of some common acoustic correlates and the measurement of an intelligibility-in-noise index.

2. The Italian language

The adaptation process used in the C2H experimental part [5] took advantage of some characteristics of the English language in which a vowel exists, [ə], which is widely recognised as the most common reduced phonetic configuration in hypo-articulated speech.

The question therefore has surged whether low-contrastive configurations could be also found in the languages, such as Italian, where low-energy phones cannot be explicitly found.

Italian is a seven-vowel language with some peculiar differences with respect to English: i) the absence of low-energy phonemes such as /ə/ and /h/ in its phonemic inventory; ii) vowel acoustic realisations mostly stand close to the border of the vocalic triangle (F1-F2 chart); iii) the high variability of stress position in the word, along with the contrastive use of it; iv) the contrastive use of consonant geminations.

Even though Italian language does not exhibit schwa in his vocalic system, it can be observed (as allophones of some unstressed vowels) in spontaneous speech, in some reduction phenomena or in some local dialects [9]. Thus, the Italian hypo-

articulated speech is also assumed to contain one (or more) low-contrastive configurations towards which vowels are reduced. The main difference with English would be the selected target phones to train the linear transformation. The contrastive use of stressed/unstressed vowels and the consonant gemination, which mostly affects the phone duration, can be also controlled to reduce the acoustic distance between close phones.

As in English, formant shifting, spectral energy redistribution, speaking rate changes, pitch modification are the most common phenomena observed in Italian hyper/hypo-articulated speech.

3. The phonetic-contrast transformation

The basic principle that drives this transformation is that low-contrastive configurations exist for both human and synthetic speech. In such configurations, produced speech is less contrastive (i.e. phones merged together) and it becomes therefore less intelligible. On the contrary, when speech production moves away from these configurations, speech becomes clearer and intelligibility increases.

The adaptation process in C2H focused on low-level signal modifications, but it was also aware of the phonetic content of speech production along with the most likely competing phones for human understanding. Low-contrastive attractors were hypothesised for every phone to define the direction for hyper/hypo-articulated speech transformation. These attractors should be the most likely acoustic realisations towards which speech production converges when the effort has to be minimised (hypo articulated speech) and from which it moves when the intelligibility has to be maximised (hyper articulated speech). When phones are reduced to these low-contrastive configurations, the acoustic distance between the most likely competing phones is minimum. An interpolation/extrapolation along the key dimension of hypo/hyper-articulation could be obtained by controlling the distance from such attractors. The proposed adaptation was achieved with a linear transformation which also allows for a continuous adaptation.

Differently from what tested in the C2H experiments on English, no distinction was made between the adaptation on vowels and the one on consonants. In both cases, it consisted on identifying a phonetically relevant competitor for each phone and assuming that the Low-Contrastive (LC) configuration is reached by applying the half-strength transformation towards it. Ideally, this technique would map both competitors into the same acoustic realisation. The High-Contrastive (HC) configuration would be achieved by moving the operational point along the same direction but opposite strength.

A small difference between vowel and consonant adaptation stood in the criterion by which the competitors were chosen. An example for vowels is displayed in Figure 1.

3.1. Implementation in an HMM-based TTS system

In order to test the proposed hyper/hypo articulated speech adaptation, the adaptation was implemented to be applied to an HMM-based speech synthesiser. Figure 2 shows the functional diagram of the procedure used to create the target corpus and to train the transformation parameters.

Starting from the set of *Full context Labels (L0)* used to build the HMM-based voice, an all-contrastive version (*L1*) was obtained through the phonetic transformation. These labels were used to generate the acoustic features (*P1*) representing the low-contrastive acoustic space. The most likely models for

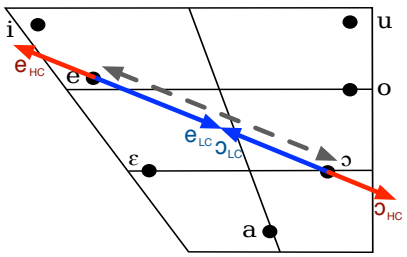


Figure 1: *Example of adaptations for [e] and [ɔ]. The blue lines refer to the transformations towards the Low-Contrastive (LC) point (hypo-articulated speech), the red lines to the ones towards the High-Contrastive (HC) (hyper-articulated). The dashed grey line shows the competitors used to train the transformation.*

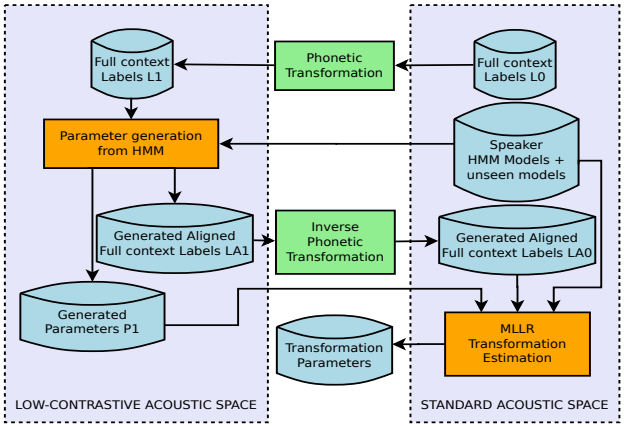


Figure 2: *Schematic diagram of the data preparation and the parameter transformation estimation.*

unseen-context phones were selected from the standard HMM models with a decision-tree clustering. The time-aligned version of *L1* (*LA1*) was mapped back into the standard phonetic domain (*LA0*). These labels along with the target *Generated Parameters (P1)* were used as reference to compute the *Transformation Parameters* with the Maximum Likelihood Linear Regression (MLLR) transformation estimation.

Once the parameters were obtained for the contrastive transformation, this would be applied with reduced strength (usually 50%) to the original HMM models to reduce standard synthesised speech to low-contrastive (hypo-articulated). Inverting the direction of the transformation as per [5], a high-contrastive (hyper-articulated) synthetic speech would be generated.

4. Italian TTS modules and voices

The base voice models used in the experiments were built using a modified version of the TTS software MaryTTS 5.0 [10]. The Italian modules and voices for MaryTTS [11] have been recently made available by the ISTC-CNR research institute and they comprehend: i) Italian lexicon and letter-to-sound rules, ii) context dependent part-of-speech tagger, iii) ToBI rules to predict symbolic prosody from text, iv) a customised version of the Italian Sampa phoneset [12].

The adaptation experiments were tested on two voice models both trained with phonetically and prosodic balanced speech corpora: *Lucia* (∼1400 sentences, ∼2 hours) recorded by a female speaker in a quasi-soundproof booth; *Roberto* (a commercial speech corpus available for research purposes, ∼1900 sentences, ∼3 hours) recorded by a professional male speaker.

Both voice models adopted MaryTTS as linguistic front-end for extracting monophone and full context labels. The HTS toolkit version 2.2 [13] was used to model the signal spectrum with Mel-Generalised Cepstral features, the fundamental frequency with multi-space probability distribution (MSD) [14], and the voicing strengths for mixed excitation [15] with continuous probability distribution.

The voices were trained with: i) default speaker-dependent HTS parameters, ii) decision tree based state clustering, iii) separate streams to model each of the static, delta and delta-delta features, iv) single Gaussian models.

Both voices are proven to have high-quality characteristics. *Lucia* is actively employed in robot-human interactions within the EU-funded project ALIZ-E[1], while the commercial voice *Roberto* has received good scores from some initial informal listening test.

5. Experiments

First a set of competing phone pairs was selected. For each vowel, the competitor was the one in opposite position across the vowel triangle (see Figure 1). On the other hand, the consonant competitor pairs were motivated by a study about perceptual confusion/discrimination on Italian consonants in noise [16]. Similarly to [17], Caldognetto listed the Italian consonants that are more likely to be mistaken in several noisy conditions. Following these guidelines, the most confusable consonant pairs were chosen to be the competitor pairs. Further contrastive pairs were motivated by the contrastive use of consonant gemination in Italian. The geminated consonants were therefore mapped into the corresponding non-geminated ones.

A summary of the substitutions is displayed in Table 1.

Table 1: *Vowel and consonant mapping. STD column contains the original phones and CTR has the contrastive ones. Geminate consonants are mapped to the corresponding non-geminate ones.*

STD	CTR	STD	CTR	STD	CTR
[a]	[u]	[f]	[p]	[dz]	[dʒ]
[e]	[o]	[t]	[k]	[dʒ]	[dz]
[i]	[ɔ]	[k]	[t]	[g]	[dʒ]
[o]	[e]	[ts]	[s]	[z]	[g]
[u]	[a]	[s]	[ts]	[l]	[ʎ]
[ɛ]	[o]	[tʃ]	[s]	[ʎ]	[l]
[ɔ]	[e]	[ʃ]	[tʃ]	[m]	[n]
[j]	[ɔ]	[b]	[d]	[n]	[m]
[w]	[a]	[d]	[b]	[ɟ]	[m]
[p]	[f]	[v]	[b]	[r]	[m]

After that, the relative adaptation for the *Lucia* and *Roberto* voices was obtained as per the method described in Section 3.1.

The same framework used to test the C2H model was also applied to evaluate this adaptation. A common test set of 200 text sentences (not included in the training set) were used to generate the test utterances which was then assessed with objective evaluations.

As mentioned above, the transformation has to be applied to the standard TTS voice (STD) with the appropriate strength in order to reach the correct low-contrastive and high-contrastive operational points. The generated speech signals were synthesized with three degree of articulation: a) HYO the lowest-contrastive configuration which is still intelligible in clean environment (60% of the total strength); b) STD the standard TTS outcome (no adaptation); c) HYP the highest-contrastive configuration which is not affected by too sever artefacts (60% of the total inverse strength).

Before the evaluation, all signals were normalised to have a constant RMS (-24 dBFS).

Thereafter, the three different kinds of speech signals were analysed using automatic tools to extract acoustic correlates and the results compared with what observed in literature for hyper and hypo-articulated speech. These speech signals were also mixed with three different disturbances: i) a real car noise, ii) a babble noise recorded in a large-size room, iii) 2-3 competing different language (English) talkers. In order to normalise the speech energy with respect to the noise, the Segmental Signal to Noise Ratio (SSNR) [18] was computed and the disturbance was amplified to have constant mean SSNR. Three SSNR levels has been taken into consideration for these experiments: 1 dB, -4 dB, and -9 dB.

6. Results

In order to evaluate the proposed adaptation performance, two types of analysis on the three types of synthesiser outcome are provided: an acoustic and an intelligibility one.

6.1. Acoustic analysis

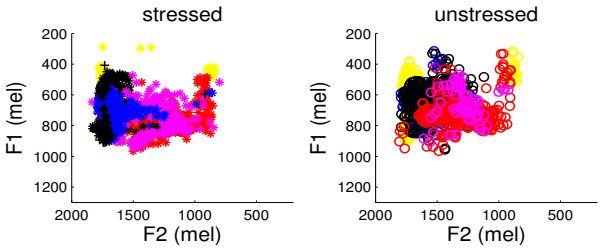
The first assessment of the adaptation effects on the acoustic signal was done by plotting the first two vowel formants (F1 and F2) of synthesiser outcome with the three degrees of articulations (HYP, HYO, and STD) (Figure 3 and Figure 4).

It is clear from these plots how both TTS voices move from the confused and centralised positions of the HYO configuration (Figure 3a and 4a) to the more separate and recognisable ones of HYP (Figure 3a and 4a). The stressed HYO vowels in *Roberto* aggregate in three main positions rather than an unique central one. This confirms the idea that the low-contrastive configuration is not an unique position close to [ə], but it is an intermediate position depending on the surrounding phones. All these behaviours emerged spontaneously from the adaptations without any assumption in the training but the control of phonetic contrast. Transformations seem to achieve a more effective reduction/expansion on the *Roberto* voice. It is worth to notice that the *Roberto* vowel variance (Figure 4) is quite limited with respect to the *Lucia* one (Figure 3). Indeed, the former was created using a professional speaker's voice whereas the latter denotes some regional accent influence. Moreover, the amount of recorded corpus is different: *Roberto* training corpus was one-third bigger the *Lucia* one.

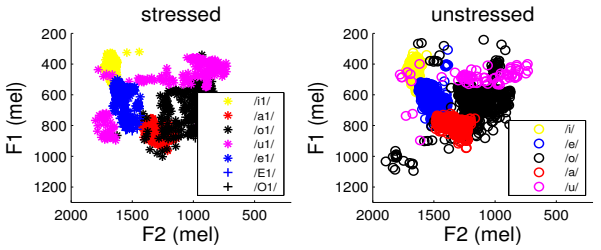
Even though two adaptations contribute to modify the signal at the same time, the vowel charts behave similarly to what observed for the schwa-based ones in [5].

Another type of acoustic analysis was performed by extracting some acoustic parameters, which are proved to be correlated to the degree of articulation of speech [19, 20], from the audio generated by the implemented adaptations:

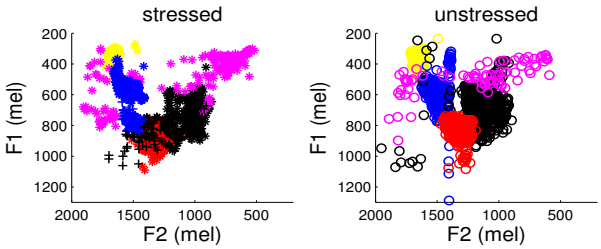
[1] http://www.aliz-e.org/



(a) HYO

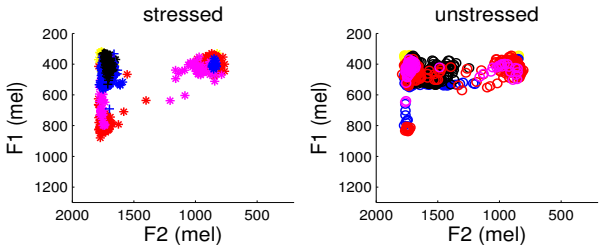


(b) STD

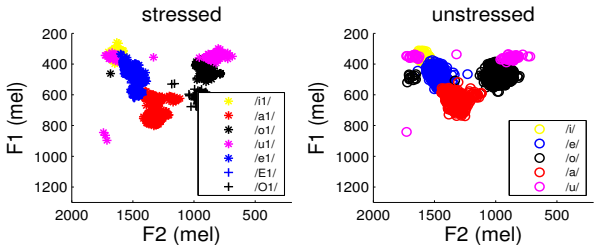


(c) HYP

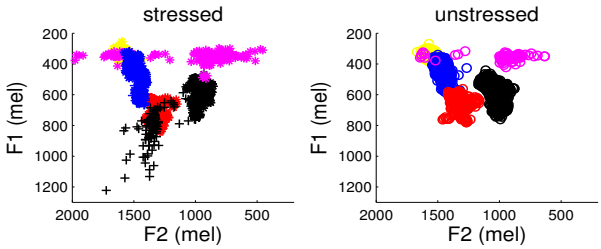
Figure 3: Effect of the *HYO* (Figure 3a) and the *HYP* (Figure 3c) adaptation applied to the *Lucia* voice. The plot for STD voice vowels is also plotted (Figure 3b) for reference. SAMPA symbols are used in the legend.



(a) HYO



(b) STD



(c) HYP

Figure 4: Effect of the *HYO* (Figure 4a) and the *HYP* (Figure 4c) adaptation applied to the *Roberto* voice. The plot for STD voice vowels is also plotted (Figure 4b) for reference. SAMPA symbols are used in the legend.

duration parameters : the Mean Sentence Duration, *MSD*, and the Mean Phone Duration (without pauses), *MPD*, which normally increase in human speech with the degree of articulation;

spectral parameters: the Long Term Average Spectrum, *LTAS13*, the spectral tilt, *Sp.Tilt*, the spectrum Centre of Gravity *Sp.CoG*, and the vowel space area (*F1F2 area*) which usually show a shifting towards high frequency;

pitch parameters: the average fundamental frequency value, *F0*, and its range, *F0 range* which should both increase accordingly to the degree of articulation.

The average result values are shown in Table 2 for *Lucia* and in Table 3 for *Roberto*. In both tables, the clearest modifications are observed in the vowel space (*F1F2 area*) expansion/reduction. Even though this was imposed by design, nonetheless this observation, together with Figure 3 and Figure 4, it proves the adaptation behaves correctly.

Other evident differences between the three sets of audio files appear in the spectrum energy shift (e.g. *Sp.CoG* and *Sp.Tilt*) and in the duration (*MSD* and *MPD*). The latter proves the tendency of the automatic system to elongate the speech production to increase phonetic contrast and vice-versa.

In conclusion, from the acoustic analysis, it can be affirmed that the proposed transforms are behaving according to what observed in human speech production.

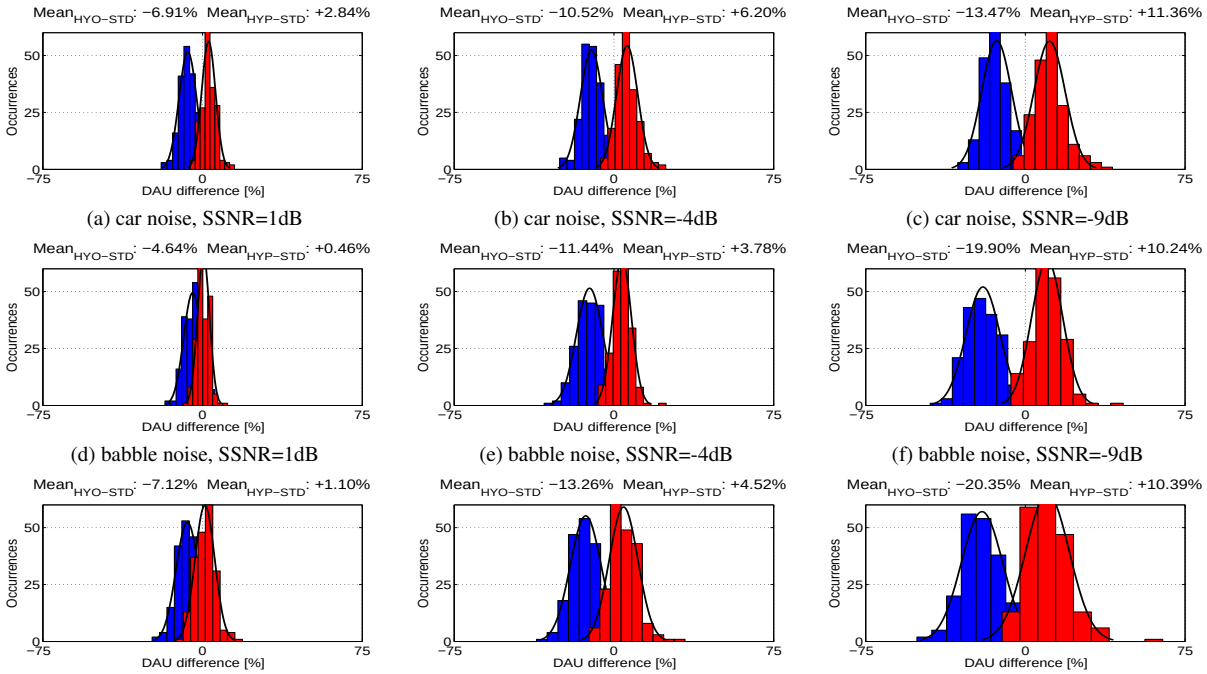
Table 2: Acoustic analysis of the three degrees of adaptation on the *Lucia* voice. In parenthesis the difference with STD.

Type of analysis	**HYO**	**STD**	**HYP**
MSD [s]	5.75 (-5.1%)	6.06	6.38 (+5.3%)
MPD [s]	0.078 (-2.5%)	0.08	0.083 (+3.7%)
LTAS13 [dB SPL]	47.7 (-9.3%)	52.6	58.3 (+10.8%)
Sp.Tilt [dB/dec]	-5.6 (+7.7%)	-5.2	-4.7 (-9.6%)
Sp.CoG [Hz]	394.1 (-27.9%)	546.2	835.9 (+53.0%)
F1F2 area [Hz^2]	14115 (-90.1%)	142401	203959 (+43.2%)
F0 [Hz]	197.3 (-3.4%)	204.3	210.3 (+2.9%)
F0 range [Hz]	138-225 (-23.6%)	133-247	134-276 (+24.8%)

6.2. Intelligibility evaluation

Intelligibility is strongly correlated with the effort involved in human speech production: the more adverse the condition, the higher the degree of articulation. Therefore, the proposed adaptation should also control the intelligibility in noisy conditions.

The HYO,HYP, and STD test files, mixed with disturbances as explained in Section 5 were processed with the Dau method [21] to assess speech intelligibility (DAU). Automatic intelligibility-assessment methods are quite important to score the speech synthesis quality. Even though most of them mainly measure the audibility of a signal without taking into account of

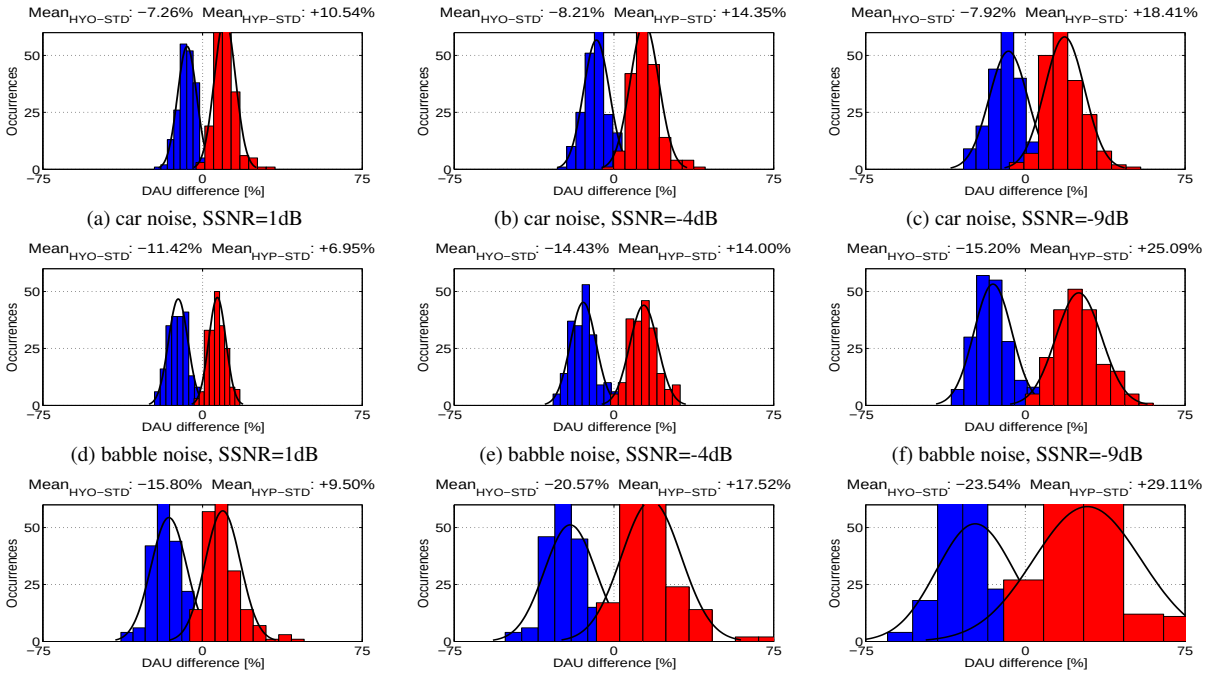


(a) car noise, SSNR=1dB (b) car noise, SSNR=-4dB (c) car noise, SSNR=-9dB

(d) babble noise, SSNR=1dB (e) babble noise, SSNR=-4dB (f) babble noise, SSNR=-9dB

(g) competing talker, SSNR=1dB (h) competing talker, SSNR=-4dB (i) competing talker, SSNR=-9dB

Figure 5: Objective evaluation of the Hyper (*HYP*) and Hypo (*HYO*) adaptation applied to the *Lucia* voice: distribution of the DAU differences (in percentage) between HYO and STD speech (blue histograms), and between HYP and STD speech (red histograms). Every row shows results for a different kind of noise and every column is related to the same SSNR.



(a) car noise, SSNR=1dB (b) car noise, SSNR=-4dB (c) car noise, SSNR=-9dB

(d) babble noise, SSNR=1dB (e) babble noise, SSNR=-4dB (f) babble noise, SSNR=-9dB

(g) competing talker, SSNR=1dB (h) competing talker, SSNR=-4dB (i) competing talker, SSNR=-9dB

Figure 6: Objective evaluation of the Hyper (*HYP*) and Hypo (*HYO*) adaptation applied to the *Roberto* voice: distribution of the DAU differences (in percentage) between HYO and STD speech (blue histograms), and between HYP and STD speech (red histograms). Every row shows results for a different kind of noise and every column is related to the same SSNR.

Table 3: Acoustic analysis of the three degrees of adaptation on the *Roberto* voice. In parenthesis the difference with STD.

Type of analysis	**HYO**	**STD**	**HYP**
MSD [s]	4.72 (-14.2%)	5.50	6.28 (+14.2%)
MPD [s]	0.06 (-16.7.5%)	0.072	0.083 (+15.3%)
LTAS13 [dB SPL]	44.7 (-6.9%)	48.0	56.3 (+17.3%)
Sp.Tilt [dB/dec]	-6.3 (+8.6%)	-5.8	-4.9 (-15.5%)
Sp.CoG [Hz]	434.5 (-30.6%)	625.8	947.0 (+51.33%)
F1F2 area [Hz^2]	469 (-99.6%)	124518	143156 (+15.0%)
F0 [Hz]	119.7 (+2.9%)	116.3	112.7 (-3.1%)
F0 range [Hz]	73-143 (-6.7%)	68-143	67-162 (+26.6%)

actual phonetic content of the signal, Dau's index is proved to be quite correlated to human understanding performances [22].

The DAU differences (in percentage) between HYO/HYP and STD are plotted in Figure 5 and 6. The figures show a clear intelligibility improvement/reduction with respect to the standard TTS voice in all types of noises for both voices. Signal modifications are more significant at medium/high levels of noise and when applied to the *Roberto* voice. On average, the intelligibility deviation from the STD voice is around 10% for both the HYO and the HYP adaptations.

7. Conclusions

In this paper, the same adaptation technique of the C2H model was applied to Italian to confirm its validity on a language with different phonological characteristics from English. A phonetic-contrast motivated transformation from standard to low-contrastive phone realisations was proposed on the basis of the vowel positions on the vocalic triangle and the perceptual discrimination of consonant pairs for this language.

The objective analyses confirm that the estimated transformation parameters can model different degrees of articulation for speech from low-contrastive (hypo-) to high-contrastive (hyper-articulated). Acoustic analyses on the generated speech confirm that the deviation of some relevant acoustic speech cues from the standard articulated speech, like vowel space expansion, frequency distribution, spectral tilt, speaking rate and pitch, follows the literature results when the hyper/hypo articulated models were used instead of the standard models. Moreover, intelligibility tests in noise condition have shown the increasing of the intelligibility of the generated hyper articulated speech with respect to the speech generated with the standard HMM models.

Hence, results indicate that the transformation, when applied to the two Italian HMM-based voices, is indeed effective, even if Italian does not have clear low-energy attractors for hypo-articulated speech in its phonetic inventory, such as schwa the for English.

8. Acknowledgements

The research leading to these results was funded by the EU-FP7 network SCALE (ITN-213850) and by EU-FP7 project ALIZ-E (ICT-248116). The authors would like to thank MiVoQ for supporting this research and providing the *Roberto* voice.

9. References

[1] É. Lombard, "Le Signe de l'Elevation de la Voix - The sign of the rise in the voice," *Ann. Maladiers Oreille, Larynx, Nez, Pharynx - Annals of diseases of the ear, larynx, nose and pharynx*, vol. 37, pp. 101–119, 1911.

[2] B. Lindblom, "Explaining phonetic variation: a sketch of the H&H theory," *Speech production and speech modelling*, vol. 55, pp. 403–439, 1990.

[3] R. K. Moore, "PRESENCE: A Human-Inspired Architecture for Speech-Based Human-Machine Interaction," *IEEE Transactions on Computers*, vol. 56, no. 9, pp. 1176–1188, Sep. 2007.

[4] R. K. Moore and M. Nicolao, "Reactive Speech Synthesis: Actively Managing Phonetic Contrast Along an H&H Continuum," in *ICPhS 2011*, Hong Kong, China, Aug. 2011, pp. 1422–1425.

[5] M. Nicolao, J. Latorre, and R. K. Moore, "C2H: A Computational Model of H&H-based Phonetic Contrast in Synthetic Speech," in *INTERSPEECH 2012*, 2012.

[6] T. Raitio, A. Suni, M. Vainio, and P. Alku, "Analysis of HMM-Based Lombard Speech Synthesis," in *INTERSPEECH 2011*, Florence, Italy, Aug. 2011, pp. 2781–2784.

[7] C. Valentini-Botinhao, J. Yamagishi, and S. King, "Mel Cepstral Coefficient Modification Based on the Glimpse Proportion Measure for Improving the Intelligibility of HMM-Generated Synthetic Speech in Noise," in *INTERSPEECH 2012*, Portland, OR, Jun. 2012, pp. 1–4.

[8] B. Picart, T. Drugman, and T. Dutoit, "Continuous control of the degree of articulation in HMM-based speech synthesis," in *INTERSPEECH 2011*, Florence, IT, 2011, pp. 1797–1800.

[9] F. A. Leoni, F. Cutugno, and R. Savy, "The vowel system of Italian connected speech," in *ICPhS 1995*, B. P. Elenius K., Ed., vol. 4, Stockholm, 1995, pp. 396–399.

[10] M. Schröder, M. Charfuelan, S. Pammi, and I. Steiner, "Open source voice creation toolkit for the MARY TTS Platform," in *INTERSPEECH 2011*, Florence, Italy, 2011.

[11] F. Tesser, G. Paci, G. Sommavilla, and P. Cosi, "A new language and a new voice for MARY-TTS," in *9th national congress, AISV (Associazione Italiana di Scienze della Voce)*, Venice, Italy, 2013.

[12] "SAMPA for Italian," 1989 (accessed May 21, 2013). [Online]. Available: http://www.phon.ucl.ac.uk/home/sampa/italian.htm

[13] H. Zen, T. Nose, J. Yamagishi, S. Sako, T. Masuko, A. Black, and K. Tokuda, "The HMM-based speech synthesis system (HTS) version 2.0," in *The 6th International Workshop on Speech Synthesis*. Citeseer, 2007, pp. 294–299.

[14] K. Tokuda, T. Masuko, N. Miyazaki, and T. Kobayashi, "Hidden Markov models based on multi-space probability distribution for pitch pattern modeling," in *ICASSP 1999*. IEEE, 1999, pp. 229–232 vol.1.

[15] T. Yoshimura, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, "Mixed Excitation for HMM-based Speech Synthesis," in *Eurospeech*, 2001.

[16] E. M. Caldognetto, K. Vagges, and F. Ferrero, "Intelligibilità e confusione consonantiche in Italiano," *Rivista Italiana di Acustica*, 1988.

[17] G. A. Miller and P. E. Nicely, "An analysis of perceptual confusions among some English consonants," *JASA*, Jan. 1955.

[18] Y. Hu and P. C. Loizou, "Evaluation of Objective Quality Measures for Speech Enhancement," *IEEE Trans. Speech, Audio & Language Processing*, vol. 16, no. 1, pp. 229–238, Jan. 2008.

[19] V. Hazan and R. E. Baker, "Acoustic-phonetic characteristics of speech produced with communicative intent to counter adverse listening conditions," *JASA*, vol. 130, no. 4, p. 2139, 2011.

[20] R. J. J. H. van Son and L. C. W. Pols, "An acoustic description of consonant reduction," *Speech Communication*, vol. 28, no. 2, pp. 125–140, Jun. 1999.

[21] T. Dau, D. Puschel, and A. Kohlrausch, "A quantitative model of the effective signal processing in the auditory system. I. Model structure," *JASA*, vol. 99, no. 6, pp. 3615–3622, Jun. 1996.

[22] C. Valentini-Botinhao, J. Yamagishi, and S. King, "Evaluation of Objective Measures for Intelligibility Prediction of Hmm-Based Synthetic Speech in Noise," in *ICASSP 2011*, Prague, May 2011.

Using neighbourhood density and selective SNR boosting to increase the intelligibility of synthetic speech in noise

Cassia Valentini-Botinhao, Mirjam Wester, Junichi Yamagishi, Simon King

The Centre for Speech Technology Research, University of Edinburgh, UK

C.Valentini-Botinhao@sms.ed.ac.uk, mwester@inf.ed.ac.uk

jyamagis@inf.ed.ac.uk, Simon.King@ed.ac.uk

Abstract

Motivated by the fact that words are not equally confusable, we explore the idea of using word-level intelligibility predictions to selectively boost the harder-to-understand words in a sentence, aiming to improve overall intelligibility in the presence of noise. First, the intelligibility of a set of words from dense and sparse phonetic neighbourhoods was evaluated in isolation. The resulting intelligibility scores were used to inform two sentence-level experiments. In the first experiment the signal-to-noise ratio of one word was boosted to the detriment of another word. Sentence intelligibility did not generally improve. The intelligibility of words in isolation and in a sentence were found to be significantly different, both in clean and in noisy conditions. For the second experiment, one word was selectively boosted while slightly attenuating all other words in the sentence. This strategy was successful for words that were poorly recognised in that particular context. However, a reliable predictor of word-in-context intelligibility remains elusive, since this involves – as our results indicate – semantic, syntactic and acoustic information about the word and the sentence.

Index Terms: word confusability, neighbourhood density, HMM-based speech synthesis

1. Introduction

Text-to-Speech (TTS) systems are deaf (and blind) to the environment and currently do not react to adverse conditions when intelligibility is possibly more important than naturalness. Although research aimed at generating clear or Lombard style synthetic speech has been carried out [1–5] such speech modification methods do not take into account word-level confusability. The best objective measures of intelligibility [6–8] are based on the acoustic and effective signal processing that takes place in the human auditory system and does not consider lexical activation or any further context information.

However, some words are inherently more intelligible than others i.e., they are less likely to be confused with other words. This property of words is currently ignored but could potentially be exploited when applying speech modifications. The premise is that modifications aimed at improving intelligibility should not necessarily be "on" the whole time. For example, we can think in terms of energy budget whereby energy in a sentence is reallocated on the basis of the expected intelligibility of a word. More or less energy is expended depending on the predicted intelligibility of a word. Another approach would be to use word confusability to control the balance between naturalness/quality of the speech and any intelligibility improvements resulting from the modification. In this case, the level of modification could be constrained by the degree of distortion it introduces. As it is not clear how to define what an acceptable amount of distortion is, we use word-level information to reallocate energy under the constraint of fixed sentence energy.

This work is a first attempt towards making use of a model of spoken word activation, the neighbourhood activation model (NAM) [9], in an energy-based speech modification. We address two questions: can neighbourhood density values be used to predict intelligibility at the word-level and can we use this information to improve overall word recognition by selectively boosting highly confusable words.

Of course there are many factors that influence the intelligibility of a word: acoustic confusability, linguistic confusability, the inherent intelligibility of a speaker, environmental factors (e.g., noise types) and listener characteristics. How to predict which words in a sentence are going to be easily intelligible and which ones hard is therefore not straightforward. Furthermore, in order to measure the effectiveness of selectively boosting words based on their intelligibility, the influence of all these different factors needs to be restricted. Therefore, we decided on the following constraints in our experiments. We only consider confusability at the word-level (no linguistic confusability), synthetic speech from one speaker and one type of noise (speech-shaped noise). The modification we are looking at is energy reallocation, which we view as a starting point for other types of modifications.

The remainder of this paper describes three listening tests. In the first experiment, we investigated the use of neighbourhood density as a predictor of word intelligibility of synthetic speech in noise for words in isolation. In the second and third experiments, the words from the first experiment were placed in matrix style sentences and energy reallocation was applied aiming for maximum intelligibility in two different ways. The following sections describe the set up of the experiments, our findings and a discussion of the results.

2. Methodology

2.1. Word selection criteria

To test different word boosting strategies it is important to select words that cover a wide range of acoustic confusability, that is easy- and hard-to-understand words. One way of performing this categorization is to use the neighbourhood density value of words. Lexical or phonological neighbourhood density (ND) plays an important role in word recognition. Words with many lexical neighbours, differing by one phoneme insertion, deletion or substitution are more difficult to recognise that words with few lexical neighbours [9]. In De Cara and Goswami [10] a second definition of phonological neighbourhood is given: the OVC-metric. In this metric, words that differ by insertions,

deletions or substitutions in either the onset, vowel or coda of a word are counted. According to the OVC-metric not only words like *main* and *gain* are phonological neighbours but also for example *main* and *strain*.

In our study, we set out to define a set of "hard" words and a set of "easy" words in terms of intelligibility. The words were selected to fill slots in Matrix-style sentences [11] of the form: [imperative verb] the [adjective] [adjective] [noun]. We chose 10 verbs, 20 adjectives and 10 nouns from an existing monosyllabic lexical database which contained both neighbourhood density statistics and frequency statistics [10]. Our criteria, similar to those used by [12], were:

- written and spoken frequency $\geq$ 10 per million,
- per Matrix slot:
  - 5 "hard" words, i.e., from a dense neighbourhood, ND-OVC $\geq$ 37,
  - 5 "easy" words, i.e., from a sparse neighbourhood, ND-OVC $\leq$ 17.

The intervals of ND-OVC values that define the easy and hard categories were as far apart as possible under the word frequency constraints.

2.2. Synthetic speech and noise material

To build the HMM-based TTS voice for this work we used read speech recordings of a British male speaker. The voice was created from a high quality average voice model which was adapted to the speaker's voice using three hours of his speech sampled at 48 kHz as described in [5]. We used a hidden semi-Markov model as the acoustic model. The isolated words were synthesised in a carrier sentence of the format: "Now we will say "pause" *word* "pause" again". They were then automatically segmented and added to noise with 200 ms initial and final lags. The speech-shaped noise was generated using recordings of a female speaker sampled at 48 kHz (similar to [13]). Different signal-to-noise ratio (SNR) values were obtained by varying the level of the speech stimuli against a constant level of speech-shaped noise (similar to [9]).

2.3. Procedure

Stimuli were presented to native British English speakers with no hearing problems over Beyerdynamic DT770 headphones in individual sound-treated booths. The experiments were run using a custom-built MATLAB software application. Each stimulus was presented once. Listeners typed what they heard, after which the following stimulus was presented. The listeners were instructed to type 'X' if they could not make out the word(s). Word accuracy rate (WAR) was calculated as the percentage of correct word transcriptions across the listeners. Homophones, for example, a response of "sea" for "see" were considered correct.

3. Listening tests and results

First a listening test of words in isolation is described. The goal of this experiment is to find the "true" intelligibility scores of words, rather than the intelligibility expectation based only on the ND values of the words. (Although one would expect them to be similar). This is followed by sentence experiments in which the goal is to improve overall sentence intelligibility by using this prior information about word intelligibility. To this end, sentence experiments using two energy reallocation strategies were investigated. In the first one, energy is taken from one word and given to another: the giver/receiver strategy. The second strategy involves reallocating energy from the whole sentence to boost one word.

3.1. Isolated word experiment design

To find which words can benefit from being presented at higher SNR levels we need to obtain word intelligibility scores at a range of different SNR values. Before the actual listening test could be performed we needed to find the range of SNR values at which to present the isolated words. The range needed to be such that "hard" words were intelligible at the highest SNR level and "easy" words unintelligible at the lowest level. A separate listening experiment involving 10 participants was carried out to find the range. On the basis of their results five SNR values were chosen: -8, -3.5, 1, 5.5 and 10 dB.

The 40 words were presented at each of the five SNR levels (200 stimuli) randomised over four blocks (50 words per block). In each block, the SNR values were ordered from low to high. Prior to the main test, listeners received a practise session presented at a mid range SNR, using 20 words from outside the test set. At the end of the test, the participants were asked to transcribe the words in clean condition, i.e. no speech-shaped noise present. 25 listeners performed the isolated word task.

3.2. Isolated word results

Figure 1 shows scatter plots of the WAR results obtained for different ND values in clean (top) and in noise at a SNR = 5.5 dB (bottom). The results show that even in the clean condition a number of words were poorly understood, achieving less than 60% WAR. Most of these words are from a dense neighbourhood, belonging to the "hard" category. Although the linear correlation between ND and WAR is quite low (-0.46 for the clean condition and -0.31 for the noisy condition) when comparing the scatter plots of the clean and noisy conditions we can see that the "easy" words are more robust to noise. That is, the dispersion towards the low WAR region caused by the presence of noise is smaller.

As each word was presented in noise at five different SNRs we are able to draw psychometric curves for each individual word. We present the curves for verbs in Figure 2. (The results for nouns and adjectives are similar but are not presented here for brevity's sake). According to the NAM model we expect words classified as "easy" to have higher WAR than "hard" words. We can see however some words do not behave as expected. For instance the verb *have* classified as an easy word is in fact less intelligible than expected and that the verb *see* is easier to recognise than expected from its ND value. This mismatch between ND values and intelligibility scores of synthetic speech in noise is not wholly unexpected as the ND does not account for noise and type of speech (TTS).

To illustrate how challenging it is to "represent" intelligibility at the word-level we use the glimpse proportion measure (GP) [6] as a reference. We calculate the GP for each of the words in the five different SNR conditions and correlate that with the subjective scores. The GP measure was shown to obtain a high correlation coefficient (up to 0.94) with subjective intelligibility scores of a male TTS voice in diverse noise conditions when both GP and WAR scores were calculated at a word-level but averaged across the different words [14]. Here the GP values however a very poorly correlated (0.44) to WAR scores calculated for individual words.

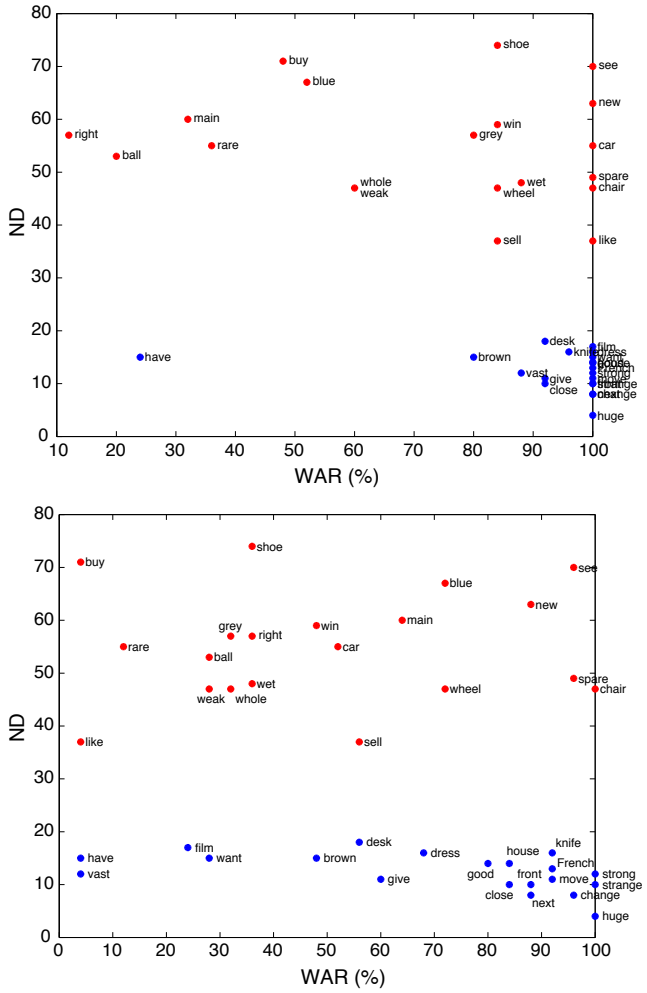


Figure 1: Neighbourhood density (ND) versus word accuracy rates (WAR) in clean (top) and in noise at a SNR = 5.5 dB (bottom), red and blue dots represent hard and easy words.

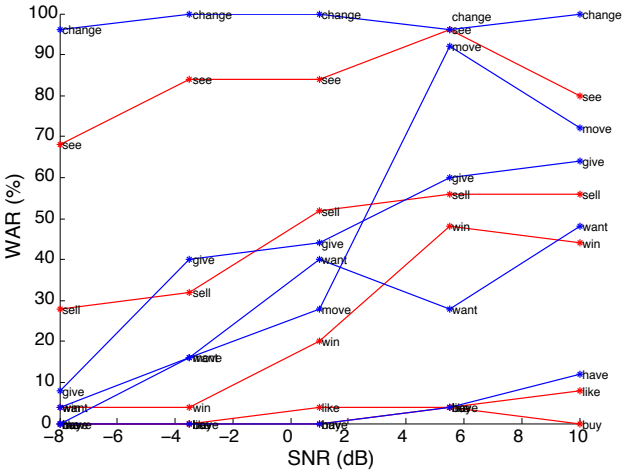


Figure 2: Psychometric curves for verbs, red and blue lines represent hard and easy words.

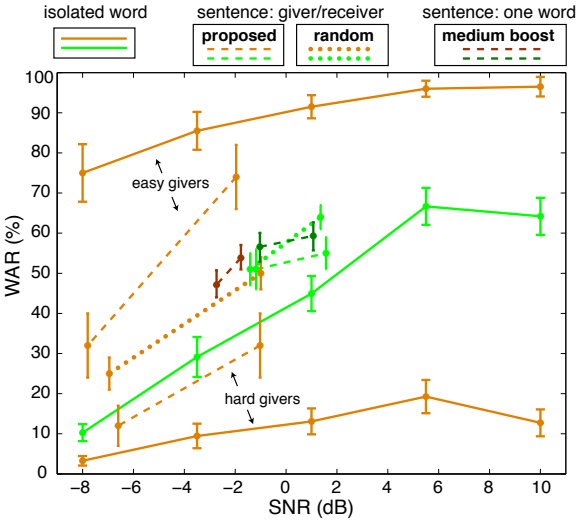


Figure 3: Psychometric curves for givers (oranges) and receivers (greens) in isolation and in sentences. "Givers" are divided into easy and hard for isolated and proposed conditions.

3.3. Sentence experiment: Giver/receiver boosting

The 40 words from the isolated word experiment were split into two categories: givers and receivers. The reallocation strategy used here is that energy is taken from one word (the giver) and given to another word (the receiver), keeping the overall energy budget the same. A word was considered a giver if the WAR in isolation for all SNRs tested as either quite low – hard giver – or quite high – easy giver. Easy and hard now relate to the words' intelligibility scores rather than their ND values. The expectation is that easy givers are robust and attenuating them will not harm their intelligibility much, whereas hard givers will remain unintelligible no matter what so they are not worth spending energy on. The receivers were words that showed steep slopes in the isolation experiment, providing evidence that at higher SNR values they were more intelligible. Our expectation is that they would benefit from energy boosting. Figure 3 shows psychometric curves for all listening tests described in this paper. Here we focus on the curves for "isolated words" – solid lines – obtained by averaging WAR values across receivers and givers. Both giver curves (easy and hard orange solid lines) are quite flat across the SNR range and on average receivers (solid green line) are more sensitive to changes in SNR, which makes them promising candidates for selective boosting.

3.3.1. Sentence material

To create the sentence material we built Matrix-style sentences of the form: [imperative verb] the [adjective] [adjective] [noun], for example: "Change the grey strange dress". Each sentence contains one giver (of energy) word and one receiver (word). The other words in the sentence (fillers) were randomly selected from the remainder of the 40 word pool, each word occurred six times. Varying the position of giver and receiver in the sentence results in 12 possible sentence types. Per sentence type, five sentences were created, resulting in 60 sentences in total. These sentences contain very little context information, aside from the structure there is no further linguistic information available to listeners, i.e., the words are equally predictable (or unpredictable).

3.3.2. Modifications

To investigate whether selectively boosting the receiver word by attenuating the giver word increases overall intelligibility we compared two types of modifications:

- **proposed** - select giver/receiver pairs according to the results of the isolated word experiment;
- **random** - select giver/receiver pairs randomly.

From the results of the isolated word experiment we expect that different receivers need different amounts of boosting to raise their WAR to a similar value, but to simplify the experiment we fix the amount of power level loss to 6 dB. On average the receivers' power increases by 2.7 dB with the constraint that the overall energy of the sentence remains unchanged.

3.3.3. Listening experiment

As words in isolation require higher SNRs to be intelligible than words in a sentence we carried out a pre-test (five participants) to find the SNR level (-3 dB) that resulted in an average of 50 % WAR across all selected sentences.

All 60 sentences were evaluated for the three different conditions: the two modifications and unmodified. As we did not want listeners to hear a sentence more than once the experiment was divided across three groups of listeners. Each listener heard all 60 sentences once and the modification type applied to each sentences was spread across the listeners so the whole test (180 stimuli) was covered by three listeners. Prior to the main test participants carried out a practice session consisting of 20 sentences of the same structure as the test however filled with other words. At the end of the test all participants were also asked to transcribe the 60 sentences in clean condition. In total, 60 listeners performed the test.

3.4. Giver/receiver sentence results

We present the average WAR of easy and hard words in Table 1 obtained in isolation and in a sentence. Easy words are more intelligible in both scenarios but the difference is less pronounced in a sentence than in isolation. These results indicate that the effect of neighbourhood density on the intelligibility of words is limited when a word is presented with context.

	easy words	hard words
isolation	93.2 (3.8)	71.2 (6.5)
sentence	97.8 (0.8)	94.4 (2.0)

Table 1: Mean word recognition (%) and its standard error for easy and hard words in isolation and in a sentence in clean conditions.

Figure 4 gives the results averaged across words and listeners for each modification in terms of absolute change compared to the unmodified case. As a reference, the rates obtained for unmodified speech were: WAR = 49.6% and for proposed/random: WAR_R = 51.25/53.3%, WAR_G = 50.5/49.9% and WAR_F = 48.3/47.6%. ($_\text{R}$ = receivers, $_\text{G}$ = givers, $_\text{F}$ = fillers). We can see that boosting a word at the detriment of another word decreases WAR results for both modifications. The intelligibility of the givers drops significantly in both cases, more than a 25% absolute drop, while the receivers only gain up to 12% in word accuracy. The results also show that on average choosing the pairs randomly rather than according to the isolated word experiment generates a larger gain for receivers

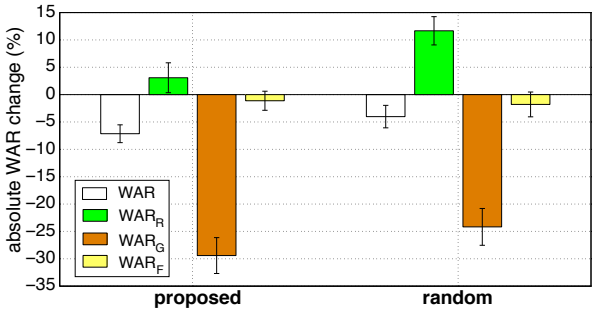


Figure 4: Absolute changes in WAR (in %) of proposed and random modifications with respect to unmodified sentences. Sentence SNR=-3 dB. $_\text{R}$ = receivers, $_\text{G}$ = givers, $_\text{F}$ = fillers.

and a smaller WAR drop for givers. A sentence-based analysis showed that the intelligibility of filler words changed significantly in some contexts even though no modification was applied to these words.

To compare the results across the two experiments, the SNR that each giver and receiver word was presented at in the sentence experiment is calculated. As the same word appears as a giver (or receiver) in more than one sentence, we obtain the word's SNR by averaging across its occurrences (as either a giver or receiver, fillers are not included here). This gives us a unique SNR per word. The results are then averaged within each category: easy givers, hard givers and receivers. These values and their standard error (which represents the variance across the words within a category) are shown in Figure 3 (sentence:giver/receiver curves) in addition to the earlier discussed results for words in isolation. Note that the sentence results only contain two points along the x-axis (SNR), because words were either boosted (receivers) or attenuated (givers), whereas in the isolated word experiment words were played at five different SNR values.

Looking at the proposed modification (dashed line) it can be seen that the hard givers and receivers are more intelligible in a sentence than in isolation (WARs: 11% to 31% hard giver; 31% to 58% receiver) whereas the easy givers are on average less intelligible in a sentence (WARs: 88% to 74%). The slopes of the curves are also different, that is, the easy and hard givers' WAR drops more than expected and receivers' WAR does not increase as much. It seems too much energy is taken from the givers while the receivers are not getting enough, which explains why the WAR per sentence does not increase.

The dotted line in Figure 3 shows the psychometric curves for randomly chosen receivers and givers (only one curve as there is no notion of easy and hard givers in the random condition). Choosing giver and receiver pairs randomly brings their psychometric curves closer to each other. Although the increase in SNR value is similar across proposed and random modifications, the intelligibility of words in the random selection improves more. This seems to be caused by the fact that words originally classified as hard givers are now in the receiver category. Basically the hard givers – words for which we expected boosting to be ineffective based on their scores in isolation– benefit most from boosting.

3.5. Sentence experiment: boosting one word

The previous experiment showed us that boosting one word and attenuating another word in the same sentence impacts on the

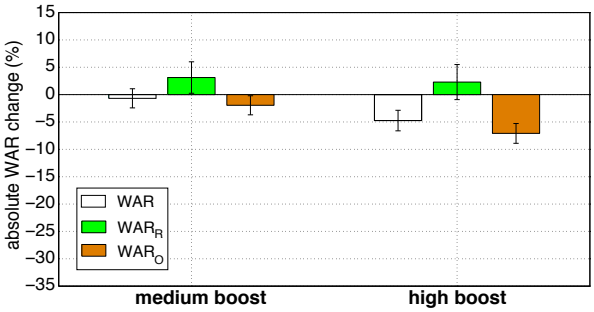


Figure 5: Absolute changes in WAR (in %) of the medium boost and high boost modifications with respect to unmodified sentences. Sentence SNR=−3 dB. $_R$ = receivers, $_O$ = others.

intelligibility of the other words in the sentence. Not only that, taking energy from one word to give to another does not improve overall intelligibility rates, mostly because the intelligibility of the attenuated word drops at a much higher rate than that the intelligibility of the boosted word increases. To overcome these two issues, we use a different type of energy reallocation strategy: energy is reallocated from the whole sentence to boost just one word. This can be viewed as emphasising a word in a sentence while making the rest of the words more quiet, possibly a slightly more natural occurring modification. Even though we saw, in the previous experiment, that randomly selecting givers and receivers resulted in higher receiver gains we have kept the same set of receiver words in this third experiment to be able to analyse the proposed selection under a more promising modification strategy.

3.5.1. Modifications

To investigate whether boosting one word in the sentence while keeping the overall SNR fixed (−3 dB) increases intelligibility we evaluate the following modifications:

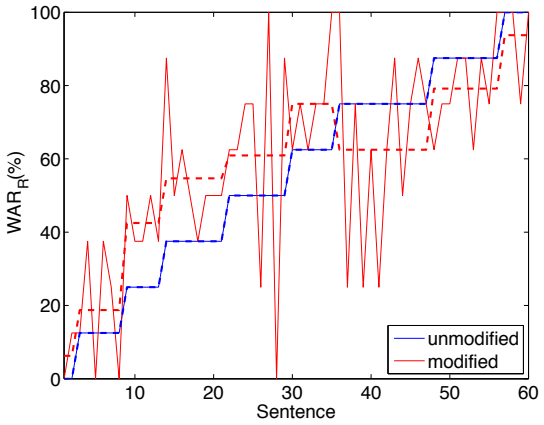
- **medium boost** - boost the receiver word by 3 dB and attenuate sentence;
- **high boost** - boost the receiver word by 5 dB and attenuate sentence.

3.5.2. Listening experiment

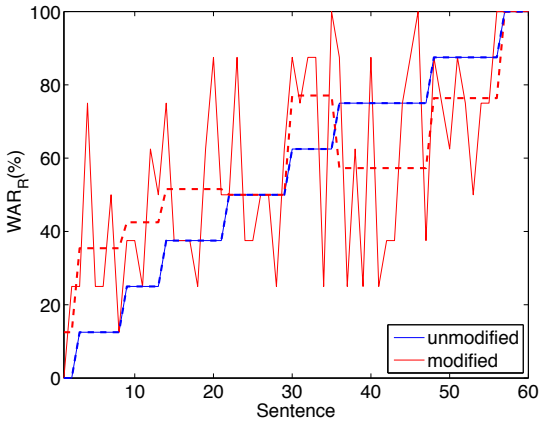
The set-up of this experiment (boosting a single word) was exactly like the giver/receiver listening test. 24 participants took part in this experiment.

3.6. Results for boosting one word experiment

Figure 5 shows WAR results averaged across words and listeners in terms of absolute change compared to the unmodified case for medium and high boost modifications. WAR_O refers to the intelligibility of the other words in the sentence –the attenuated words. As a reference, the WAR values obtained for unmodified speech were: WAR = 54.5 %, WAR_R = 56.0 % and WAR_O = 53.9 %. Although there is still an overall drop in intelligibility the drop is much smaller than what was observed in the previous experiment, see Figure 4. Particularly, we note that the medium boost modification WAR_R gains are comparable to the ones obtained using the proposed modification in the previous experiment (around 3.0 % absolute gain). However, the loss in WAR_O is much smaller (from 29.0 % to 1.9 % absolute drop)



(a) Medium boost



(b) High boost

Figure 6: Receivers' word accuracy rate (WAR_R) for each of the 60 sentences. The sentence index is ordered according to the unmodified scores (blue). Modified scores are presented at the sentence level (red continuous line) and the sentence interval level (red dashed line). Sentence SNR=−3 dB.

indicating that boosting one word in a sentence is a much better strategy.

Figure 3 shows the psychometric curves for receivers and others in the medium boosting condition (sentence: one word curves). We can see that the slope for receivers (dark green dashed line) for the medium boost modification is similar to the slope for the proposed modification (light green dashed line) and that the slope for others (i.e. givers, dark orange dashed line) is similar to the easy givers (light orange dashed line).

To identify under which conditions there was an increase of word intelligibility Figure 6 presents a sentence level analysis: WAR_R results for each of the 60 sentences. The sentences are ordered according to the WAR_R obtained in the unmodified case to check how this affect the results. The continuous red curve represents the WAR_R for the medium boost (top) and the high boost (bottom) modifications. The dashed red curves represent these results averaged across each sentence interval. A sentence interval is taken as the range where unmodified WAR_R results are constant. It can be seen that for highly intelligible words boosting can decrease the WAR_R, for both medium and high boost modifications. It seems that if a word is more intelligible than a certain threshold boosting is harmful and that this thresh-

old depends on the level of boosting. We can also see that the effect of the boosting value depends on the WAR of the receiver: poor receivers should be boosted more and highly intelligible receivers should not be boosted at all. The best strategy is then to boost the most unintelligible words in the sentence and apply an energy boost inversely proportional to the intelligibility of the word.

4. Discussion and Conclusions

This study aimed to capitalise on the idea that modifications should not be "on" the whole time but rather applied more judiciously to enhance intelligibility of synthetic speech in noise. Our experiments were designed to constrain the factors that influence the predictability of words to acoustic level confusability, to enable us to measure the effectiveness of boosting words based on their intelligibility. We carried out an isolated word experiment with 20 "hard" words from a dense and 20 "easy" words from a sparse neighbourhood according to the OVC metric in order to cover a wide range of confusability. However, our results showed that not only does neighbourhood density (ND) affect the intelligibility of words in isolation but the type of speech –a TTS voice– , the noise –speech-shaped noise– and the lexical complexity of each word also influence a words' intelligibility. Therefore, instead of using ND to select words to boost and attenuate we used the actual subjective intelligibility scores of words in isolation.

Two sentence experiments were performed using a set of 60 Matrix-style sentences which were created using the 40 easy and hard words. In the first experiment, the modification strategy was to boost one word –the receiver– by 2.7dB while attenuating another –the giver– by 6dB. The results showed that boosting a word to the detriment of another is not a good strategy, independent of the selection of the words: the intelligibility of the giver word drops by 30% absolute while receivers only increase by 3%. Moreover selecting word pairs according to their intelligibility scores in isolation performed worse than selecting them randomly. The psychometric curves for intelligibility of words in speech-shaped noise change significantly when going from isolation to a sentence, both in terms of offset and slope. Intelligibility scores of words in isolation are a poor predictor of intelligibility scores in a sentence. Furthermore, the intelligibility of words whose energy remained unmodified also changed, showing the giver/receiver strategy is not an appropriate strategy.

In the second sentence experiment, the modification strategy was to boost one word while attenuating all the other words in the sentence. The results show that this is a better modification strategy as the decrease in intelligibility for givers went from 30% to only 3%. Spreading the attenuation across all other words in a sentence is beneficial as well as being more natural. The overall gains in the intelligibility of the receiver words was still limited. Analysis at a sentence level showed that boosting is most beneficial when the intelligibility of a word is poor and the boosting level is appropriate. Boosting becomes harmful when the intelligibility of the word is already reasonable to good.

Selectively boosting words by reallocation of energy can be useful for improving intelligibility in noise, but only if the word is poorly understood to start with. If we have reliable ways of predicting word-level intelligibility it is possible to increase sentence intelligibility by selectively boosting the energy of a highly confusable word. This is a promising result that advocates the use of word intelligibility scores as prior knowledge for more complex modifications. The poor word-level intelligibility prediction results using the neighbourhood density and the glimpse proportion measure indicate however that much work needs to be done in order to obtain reliable measures of word-level confusability even for the simplest scenario of words in isolation. Translating that to sentences is an even larger challenge. The fact that subjective scores of words in isolation hardly reflect their scores in a sentence indicates that this prediction has to consider the context of the word in a sentence, not only for the additional linguistic cues but also for the acoustic coarticulation cues as well.

5. Acknowledgement

The research leading to these results was partly funded from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement 213850 (SCALE) and 256230 (LISTA), and from EPSRC grants EP/I031022/1 (NST) and EP/J002526/1 (CAF).

6. References

[1] B. Langner and A. W. Black, "Improving the understandability of speech synthesis by modeling speech in noise," in *Proc. ICASSP*, vol. 1, 18-23, 2005, pp. 265–268.

[2] T. Raitio, A. Suni, M. Vainio, and P. Alku, "Analysis of HMM-based lombard speech synthesis," in *Proc. Interspeech*, Florence, Italy, August 2011.

[3] B. Picart, T. Drugman, and T. Dutoit, "Continuous control of the degree of articulation in HMM based speech synthesis," in *Proc. Interspeech*, Florence, Italy, 2011.

[4] M. Nicolao, J. Latorre, and R. K. Moore, "C2H A computational model of H&H-based phonetic contrast in synthetic speech," in *Proc. Interspeech*, Portland, USA, September 2012.

[5] C. Valentini-Botinhao, J. Yamagishi, and S. King, "Mel cepstral coefficient modification based on the Glimpse Proportion measure for improving the intelligibility of HMM-generated synthetic speech in noise," in *Proc. Interspeech*, Portland, USA, September 2012.

[6] M. Cooke, "A glimpsing model of speech perception in noise," *J. Acoust. Soc. Am.*, vol. 119, no. 3, pp. 1562–1573, 2006.

[7] C. Christiansen, M. S. Pedersen, and T. Dau, "Prediction of speech intelligibility based on an auditory preprocessing model," *Speech Comm.*, vol. 52, no. 7-8, pp. 678–692, 2010.

[8] C. H. Taal, R. C. Hendriks, R. Heusdens, and J. Jensen, "A short-time objective intelligibility measure for time-frequency weighted noisy speech," in *Proc. ICASSP*, Dallas, USA, March 2010, pp. 4214–4217.

[9] P. Luce and D. Pisoni, "Recognizing spoken words: The neighborhood activation model," *Ear and hearing*, vol. 19, no. 1, pp. 1–36, 1998.

[10] B. Cara and U. Goswami, "Similarity relations among spoken words: The special status of rimes in English," *Behavior Research Methods, Instruments and Computers*, vol. 34, pp. 416–423, 2002.

[11] W. A. Dreschler, "Hearing in the communication society D-2-2 deliverable," 2006. [Online]. Available: http://hearcom.eu

[12] M. Cooke, "Discovering consistent word confusions in noise," in *Proc. Interspeech*, Brighton, U.K., 2009.

[13] M. Cooke, C. Mayo, C. Valentini-Botinhao, Y. Stylianou, B. Sauert, and Y. Tang, "Evaluating the intelligibility benefit of speech modifications in known noise conditions," *Speech Comm.*, vol. submitted, 2012.

[14] C. Valentini-Botinhao, J. Yamagishi, and S. King, "Can objective measures predict the intelligibility of modified HMM-based synthetic speech in noise?" in *Proc. Interspeech*, Florence, Italy, August 2011.

Noise Robustness in HMM-TTS Speaker Adaptation

Kayoko Yanagisawa, Javier Latorre, Vincent Wan, Mark J. F. Gales and Simon King

Toshiba Research Europe Ltd., Cambridge Research Lab, 208 Science Park, Cambridge, UK
{kayoko.yanagisawa,javier.latorre,vincent.wan,mjfg}@crl.toshiba.co.uk,Simon.King@ed.ac.uk

Abstract

Speaker adaptation for TTS applications has been receiving more attention in recent years for applications such as voice customisation or voice banking. If these applications are offered as an Internet service, there is no control on the quality of the data that can be collected. It can be noisy with people talking in the background or recorded in a reverberant environment. This makes the adaptation more difficult. This paper explores the effect of different levels of additive and convolutional noise on speaker adaptation techniques based on cluster adaptive training (CAT) and average voice model (AVM). The results indicate that although both techniques suffer degradation to some extent, CAT is in general more robust than AVM.

Index Terms: speech synthesis, cluster adaptive training, speaker adaptation, average voice models, noise robust adaptation

1. Introduction

With the arrival of smartphones and tablets, text-to-speech (TTS) systems are becoming more and more ubiquitous. However, most are still limited in the number of voices and/or expressions they can provide. For users of TTS applications such as Augmentative and Alternative Communication (AAC) devices, the ability to be uniquely identified by their own voice is an important aspect. For more casual users of TTS, too, personalisation can add value to the TTS system.

Building a good quality speaker-dependent TTS voice requires a large database of recordings with good phonetic coverage made in a controlled environment. This is not a realistic scenario for personalisation with dysarthric patients or with a casual user of TTS. Speaker adaptation techniques for Hidden Markov model based TTS (HMM-TTS) have emerged in recent years, which allow the creation of a target speaker's voice using a small amount of speech [1]. An average voice model (AVM) is trained on a large corpus containing multiple speakers. It is then adapted to a voice using the target speaker's adaptation data. They found that six minutes of adaptation data was enough to build a voice that sounds more natural than that of a speaker-dependent system trained on thirty minutes of speech.

The ability to create one's own voice with a small amount of data opens up the possibility to offer voice personalisation capabilities to a wider public. For example, an online custom voice building service could be envisaged where users submit a small number of sentences recorded at home, which are used to adapt a pre-built model to create their voice. In that scenario, the adaptation data is likely to be recorded in an uncontrolled environment, so the data might contain background noise, reverberation, different channel effects due to the use of non-professional recording equipment, and/or various signal processing applied by the sound card. Each of these factors has a strong impact on the quality of the models that can be obtained. In order to deal with these problems, robustness to noise is a requirement for speaker adaptation systems.

Noise robustness is a well known topic in the field of automatic speech recognition (ASR) but relatively new for TTS. The effect of creating AVMs from 'noisy' ASR data was investigated in [2, 3]. The ultimate goal of that work was to produce models that could be shared by the ASR and the TTS engine of a speech-to-speech translation system. For that purpose, the effect of training TTS models on noisy ASR data was investigated. The results showed degradation with respect to training models on clean speech. However, TTS and ASR models do not need to be shared in most cases, which means that TTS systems can be trained on reasonably good data. However, the problem regarding the quality of adaptation data remains.

Some of the noise in the adaptation data can be reduced by signal processing. For example, pops can be reduced with a high-pass filter and background noise can be reduced using spectral subtraction. More sophisticated techniques were applied in [4]. These techniques can be expected to improve the adaptation outcome but there is a limit to the amount and type of noise that can be removed. This poses a problem for AVMs because the strong adaptation capability of CMLLR transforms will treat the remaining noise as part of the speaker's voice.

Multiple linear regressions systems, such as Cluster Adaptive Training (CAT) [5], Multiple-regression HSMM (MRHSMM) [6] or eigenvoices [7] also allow speaker adaptation. In these techniques, adaptation data is projected into a linear space trained on clean data. The idea is similar to the signal-subspace approach proposed for speech enhancement [8]. As the number of parameters needed by these systems is much smaller than for AVM, their adaptation capability is much weaker, especially with large amounts of adaptation data. However, this also makes them more robust with sparse adaptation data. Moreover, the speaker space corresponds to clean speech so they may be more robust to noise.

This paper studies the effect of adapting AVM and CAT models to data with different levels of background noise and reverberation. Section 2 reviews and compares CAT and AVM adaptation. Section 3 describes the experiments. Section 4 analyses speaker similarity. Section 5 concludes.

2. Cluster adaptive training

The main characteristic of CAT [9] is that the means of the distributions are linear combinations of the mean vectors of two or more clusters. In such a model, the emission probability of an observation vector for a given speaker s, and component m is

$$p(\boldsymbol{o}(t) \mid m, s, \mathcal{M}) = \mathcal{N}\left(\boldsymbol{o}(t); \boldsymbol{\mu}_m^{(s)}, \boldsymbol{\Sigma}_{v(m)}\right) \qquad (1)$$

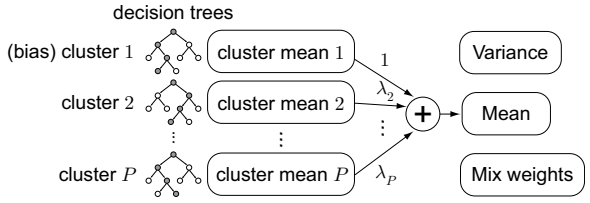


Figure 1: CAT with cluster-dependent decision trees.

with

$$\boldsymbol{\mu}_m^{(s)} = \boldsymbol{\mu}_{c(m,1)} + \boldsymbol{M}_m \boldsymbol{\lambda}_{q(m)}^{(s)} \tag{2}$$

$$\boldsymbol{\lambda}_{q(m)}^{(s)} = \left[\lambda_{2,q(m)}^{(s)}, \ldots, \lambda_{P,q(m)}^{(s)}\right]^\top \tag{3}$$

$$\boldsymbol{M}_m = \left[\boldsymbol{\mu}_{c(m,2)}, \ldots, \boldsymbol{\mu}_{c(m,P)}\right] \tag{4}$$

where $t \in \{1, \ldots, T\}$, $m \in \{1, \ldots, M\}$ and $s \in \{1, \ldots, S\}$ enumerate the frames, Gaussian components and speakers respectively; $q(m) \in \{1, \ldots, Q\}$ and $v(m) \in \{1, \ldots, V\}$ are respectively the m^{th} component's CAT regression classes and leaf node in the covariance matrices' decision tree; $c(m,i) \in \{1, \ldots, N\}$ is the leaf node for cluster i of component m in decision trees for cluster mean vectors; P is the number of clusters; $\boldsymbol{o}(t)$ is the observation vector at frame t; $\lambda_{i,q}^{(s)}$ and $\boldsymbol{\lambda}_q^{(s)}$ are respectively the i^{th} cluster's CAT weight and the weight vectors for speaker s associated with CAT regression class q; $\boldsymbol{\mu}_n$ is the cluster mean vector associated with leaf node n; $\boldsymbol{M}_m$ is component m's matrix of cluster mean vectors; $\boldsymbol{\Sigma}_k$ is leaf node k's covariance matrix; $\mathcal{M}$ is the full set of model parameters.

When each cluster is allowed its own decision tree, the result is a multi-tree model with tree-intersection as depicted in Figure 1. The main advantage of this model is its capacity to model a large number of contexts with a reduced number of parameters. An important characteristic is that the weight of the first (bias) cluster is always 1. The reason is that the goal of the bias cluster is to model those attributes which are common to all speakers. Covariance matrices and priors for the multi-space distributions (MSD) [10] could have their own tying structures, but usually they share the decision trees of the bias cluster.

The auxiliary function of the EM algorithm for the distribution of (1) is

$$\mathcal{Q}(\mathcal{M}; \hat{\mathcal{M}}) = -\frac{1}{2} \sum_{m,t,s} \gamma_m(t,s) \times \left\{ \left(\boldsymbol{o}(t) - \boldsymbol{\mu}_m^{(s)}\right)^\top \boldsymbol{\Sigma}_{v(m)}^{-1} \left(\boldsymbol{o}(t) - \boldsymbol{\mu}_m^{(s)}\right) + \log\left|\boldsymbol{\Sigma}_{v(m)}\right| \right\} + C \tag{5}$$

where C is a constant, $\hat{\mathcal{M}}$ is the current estimate of $\mathcal{M}$, and $\gamma_m(t,s)$ is the posterior probability of component m generating $\boldsymbol{o}(t)$ given s and $\hat{\mathcal{M}}$. Maximising $\mathcal{Q}(\mathcal{M}; \hat{\mathcal{M}})$ w.r.t. $\boldsymbol{\mu}_n$ yields

$$\hat{\boldsymbol{\mu}}_n = \boldsymbol{G}_{nn}^{-1} \left(\boldsymbol{k}_n - \sum_{\nu \neq n} \boldsymbol{G}_{n\nu} \boldsymbol{\mu}_\nu \right) \tag{6}$$

where

$$\boldsymbol{G}_{n\nu} = \sum_{\substack{m,i,j \\ c(m,i)=n \\ c(m,j)=\nu}} \boldsymbol{G}_{ij}^{(m)}, \qquad \boldsymbol{k}_n = \sum_{\substack{m,i \\ c(m,i)=n}} \boldsymbol{k}_i^{(m)} \tag{7}$$

and $\boldsymbol{G}_{ij}^{(m)}$ and $\boldsymbol{k}_i^{(m)}$ are accumulated statistics defined as

$$\boldsymbol{G}_{ij}^{(m)} = \sum_{t,s} \gamma_m(t,s) \lambda_{i,q(m)}^{(s)} \boldsymbol{\Sigma}_{v(m)}^{-1} \lambda_{j,q(m)}^{(s)} \tag{8}$$

$$\boldsymbol{k}_i^{(m)} = \sum_{t,s} \gamma_m(t,s) \lambda_{i,q(m)}^{(s)} \boldsymbol{\Sigma}_{v(m)}^{-1} \boldsymbol{o}(t). \tag{9}$$

By combining (6) for all the mean vectors, the update equations can be written as

$$\begin{bmatrix} \boldsymbol{G}_{11} & \ldots & \boldsymbol{G}_{1N} \\ \vdots & \ddots & \vdots \\ \boldsymbol{G}_{N1} & \ldots & \boldsymbol{G}_{NN} \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{\mu}}_1 \\ \vdots \\ \hat{\boldsymbol{\mu}}_N \end{bmatrix} = \begin{bmatrix} \boldsymbol{k}_1 \\ \vdots \\ \boldsymbol{k}_N \end{bmatrix}. \tag{10}$$

The order of (10) can be very large but it is a sparse optimisation problem because for most leaves $\boldsymbol{G}_{n,\nu} = \boldsymbol{0}$. Furthermore, if the covariances are diagonal, each dimension can be solved independently. The update equations for $\boldsymbol{\Sigma}_k$ and $\boldsymbol{\lambda}_q^{(s)}$ are

$$\hat{\boldsymbol{\Sigma}}_k = \frac{\sum_{\substack{t,s,m \\ v(m)=k}} \gamma_m(t,s) \left(\boldsymbol{o}(t) - \boldsymbol{\mu}_m^{(s)}\right) \left(\boldsymbol{o}(t) - \boldsymbol{\mu}_m^{(s)}\right)^\top}{\sum_{\substack{t,s,m \\ v(m)=k}} \gamma_m(t,s)} \tag{11}$$

$$\boldsymbol{\lambda}_q^{(s)} = \Big(\sum_{\substack{t,m \\ q(m)=q}} \gamma_m(t,s) \boldsymbol{M}_m^\top \boldsymbol{\Sigma}_{v(m)}^{-1} \boldsymbol{M}_m \Big)^{-1} \sum_{\substack{t,m \\ q(m)=q}} \gamma_m(t,s) \boldsymbol{M}_m^\top \boldsymbol{\Sigma}_{v(m)}^{-1} \boldsymbol{o}(t). \tag{12}$$

2.1. Tree building

Tree building in a tree-intersection model is computationally expensive [11]. To solve this, a cluster by cluster approach is used [12] in which the tree for one cluster is updated while the trees of the other clusters and their canonical parameters are held fixed. As usual, each tree is built to maximise the log-likelihood given the training data. Following [13], the log-likelihood for the n^{th} node in the i^{th} cluster can be computed as

$$\mathcal{L}(n) = \frac{1}{2} \hat{\boldsymbol{\mu}}_n^\top \left(\sum_{m \in \mathcal{S}(n)} \boldsymbol{G}_{ii}^{(m)} \right) \hat{\boldsymbol{\mu}}_n \tag{13}$$

with $\hat{\boldsymbol{\mu}}_n$ the ML estimate of $\boldsymbol{\mu}_n$ which is

$$\hat{\boldsymbol{\mu}}_n = \left(\sum_{m \in \mathcal{S}(n)} \boldsymbol{G}_{ii}^{(m)} \right)^{-1} \times \sum_{m \in \mathcal{S}(n)} \left(\boldsymbol{k}_i^{(m)} - \sum_{j \neq i} \boldsymbol{G}_{ij}^{(m)} \boldsymbol{\mu}_{c(m,j)} \right). \tag{14}$$

For each node n, the optimum split question q is the one that maximises the log-likelihood gain

$$\mathcal{L}(n; q) = \mathcal{L}(n_+^q) + \mathcal{L}(n_-^q) - \mathcal{L}(n). \tag{15}$$

In this way, the best question to split the n^{th} node can be selected based on the log-likelihood gain. The splitting process is stopped when a reasonable balance between complexity and accuracy is achieved. In the experiments in Section 3, minimum description length (MDL) [14] was used. After constructing the decision trees for a cluster, decision trees for the next cluster are re-built in the same manner. This process is repeated from cluster 1 to P, and the whole process repeated as desired.

2.2. CAT vs. AVM

In CAT, speaker variety is captured by the weight vector $\boldsymbol{\lambda}^{(s)}$. This vector may be interpreted as a point in eigenspace representing all possible speakers. The space is spanned by the bases defined by the CAT clusters. Since the CAT model is trained on clean speech, this speaker space is expected to be clean also. Given that CAT adaptation estimates only the $\boldsymbol{\lambda}^{(s)}$, there are insufficient degrees of freedom to capture noise in the adaptation data. Therefore CAT adaptation is constrained to yield (mostly) clean synthesis.

In contrast, the emission probability for a given component and speaker using CMLLR or CSMAPLR transforms is

$$p(\boldsymbol{o}(t) \mid m, s, \mathcal{M}) = |\boldsymbol{A}^{(s)}_{r(m)}| \mathcal{N}\left(\boldsymbol{A}^{(s)}_{r(m)}\boldsymbol{o}(t) + \boldsymbol{b}^{(s)}_{r(m)}; \boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m\right) \tag{16}$$

where $r(m) \in \{1, \ldots, R\}$ is the regression class associated with component m and $\{\boldsymbol{A}^{(s)}_{r(m)}, \boldsymbol{b}^{(s)}_{r(m)}\}$ the CMLLR transform associated with class $r(m)$. Comparing (16) and (1) it is obvious that a CMLLR transform is much more powerful than the CAT weights, allowing not just translations but also rotations of the acoustic space. This allows AVM to produce better speaker similarity than CAT when there is sufficient adaptation data. However, when the adaptation data is sparse, some of the transforms can not be estimated robustly. Therefore, both the similarity and the quality degrades [15]. In the case of adaptation with noisy data, this extra freedom might also be problematic. CMLLR has no mechanism to constrain the adapted models within a sub-space of "clean" speech. Thus it is likely to treat the noise as an attribute of the speaker.

3. Experiments

3.1. Data

Speech data from a variety of sources were used for training the models. They consisted of a) high quality recordings of phonetically balanced sentences read by professional voice talents with a neutral style in specialist recording studios; b) cheaper, lower quality studio recordings made in less strictly controlled conditions; and c) amateur-read audiobooks which were published freely on the Internet. In total, 20 speakers provided just under 30 hours of data and they all spoke US English with the General American accent.

A separate test set consisting of 8 male and 8 female non-professional speakers was recorded for adaptation. The recordings were of neutrally read sentences made in quiet office rooms using a headset microphone on a laptop with all signal processing effects turned off. Each speaker spoke the same set of 100 sentences, amounting to about 7 minutes of speech per speaker. All speakers spoke US English but not strictly the General American accent.

3.1.1. Simulation of noise

The clean data was corrupted with additive and convolutional noise to simulate noisy adaptation data. Multi-speaker babble consisting of real world multi-talker non-stationary environment noise captured at a trade show was used as additive noise. This was added to the adaptation data at signal-to-noise ratios (SNRs) of 0dB (*BAB00*) and 5dB (*BAB05*).

Convolutional noise was simulated by adding reverberation to the signal using the reverb effect in the digital audio editor, SoX [16]. The data was corrupted with two levels of reverberation: 30% (*RVB30*) and 60% (*RVB60*). The percentages indicate the proportion of output signal occupied by reverberation.

3.1.2. Pre-processing

A way to overcome the problems of noisy adaptation data is to apply signal pre-processing. In a standard custom voice building scenario, a certain amount of background noise is expected in the adaptation data, as well as pops produced by recording with the microphone directly in the airstream. Therefore a pre-processing scheme of silence trimming, high-pass filtering, spectral subtraction and amplitude normalisation was devised.

Pilot experiments showed that with babble and clean data, pre-processing improved the quality of output speech for both CAT and AVM-adapted models. However, with reverberation, there was no significant preference for CAT, and non pre-processed data was preferred for AVM. This could be explained by the fact that the type of pre-processing applied aims to remove additive noise and pops but does not deal with convolutional noise. Therefore, in these experiments, pre-processing was applied for the babble conditions but not for the reverb conditions.

3.2. Parameterisation and label generation

Waveforms were down-sampled to 22050Hz. They were then parameterised using 40 dimensional Mel-LSP coefficients with deltas, log-F0 with first and second order deltas and 20 linear-scale band aperiodicities with deltas. Context feature labels were generated automatically on the clean data.

3.3. Models

3.3.1. AVM model build

The AVM model employs CMLLR and CSMAPLR transforms [1]. A standard training procedure was used: A speaker-independent monophone maximum likelihood model is built and then CMLLR speaker adaptive training is applied. The monophone models are cloned to full context models which are clustered using decision trees. Speaker adaptive training continues with block diagonal global CMLLR transforms for speech, silence and pause. The decision trees, canonical model and global CMLLR transforms are updated several times iteratively. Regression class CMLLR transforms are then trained with the decision trees held fixed and the model parameters updated. The state-duration distributions are treated similarly.

To synthesise a new voice, some samples of the target speaker (adaptation data) are used to create an initial CMLLR transform which is then refined using CSMAPLR followed by a speaker-dependent MAP adaptation of the means so that

$$\hat{\boldsymbol{\mu}}^{(s)}_m = \frac{\tau\boldsymbol{\mu}_m + \sum_{t\in t(s)} \gamma_m(t,s)\left(\hat{\boldsymbol{A}}^{(s)}_{r(m)}\boldsymbol{o}(t) + \hat{\boldsymbol{b}}^{(s)}_{r(m)}\right)}{\tau + \sum_{t\in t(s)} \gamma_m(t,s)} \tag{17}$$

where $\boldsymbol{\mu}_m$ is the mean vector of the AVM; $\gamma_m(t,s)$, $\{\hat{\boldsymbol{A}}^{(s)}_{r(m)}, \hat{\boldsymbol{b}}^{(s)}_{r(m)}\}$ and $t(s)$ are the state occupancy probability, CSMAPLR transforms and data for speaker s respectively, and τ the hyperparameter. The MAP adapted model is combined with the CSMAPLR transform for synthesis.

Potentially, by adding extra freedom, a MAP update of the means may be more susceptible to noisy adaptation data. In initial tests, however, AVM synthesis with and without the final MAP update did not produce noticeably different samples.

3.3.2. CAT model build

A CAT model with six clusters and a bias was built as follows. The AVM canonical model was converted into a speaker-independent model by running one update to remove the CMLLR transforms. This speaker-independent model is copied into the bias cluster. The six additional clusters were initialised with zero means. Each training speaker is assigned to one of the six clusters by perceived similarity[1]. The initial CAT weights were set to 1/0 values corresponding to the speaker's assigned cluster and a value of 1 for the bias. In this way, given (2), the initial CAT model is effectively identical to the speaker-independent model. MDL based decision tree context clustering was performed for each cluster leaving the bias cluster until last. The aim at this stage is to coax the model in such a way that each cluster models speaker specific attributes while the bias cluster models common attributes. Alternative initialisation schemes may be envisaged (e.g. [15]). After context clustering is performed for all CAT clusters, the model's parameters (means and variances) and CAT weights are updated iteratively. Note that the weights are updated independently for each speaker in the training set. The initial grouping of the speakers merely provide a starting point and does not tie the weights of different speakers. The context clustering and iterative model/CAT weight updates are repeated once.

To synthesise a new speaker, an initial set of CAT weights are copied from one of the training speakers. The weights are updated iteratively to maximise the likelihood given the adaptation data. It was observed that the weights converge to the same values irrespective of the starting point.

3.4. Evaluation setup

In order to avoid a walkie-talkie effect resulting from noise being modelled in the start and end silences, CAT weights, CSMAPLR transforms and models for silence were replaced, post-adaptation, with those obtained using clean data. Speech waveforms were synthesised from the generated speech parameters with post-filtering.

Subjective listening tests were conducted via the crowdsourcing website *CrowdFlower* using Mechanical Turk workers located in the US [17]. Listeners were asked to rate the quality of the synthetic speech on a five-point scale where 1 is very bad and 5 is very good. The top end of the scale was anchored with natural speech samples from the same speakers. To anchor the bottom end of the scale, speaker-dependent models were trained for each speaker, with the standard HMM-TTS flat-start approach using the 100 adaptation sentences only.

3.5. Results

Synthesis of the CAT model adapted to noisy data still yielded speech that was clean but with slightly degraded quality. In contrast the AVM's synthesis was corrupted by noise with properties resembling the noise contained in the adaptation data. These observations are consistent with the theory.

Results of the MOS tests are shown in Figure 2. CAT adaptation outperforms the AVM for *CLEAN*, as expected from previous studies [15]. In addition, it is more robust to noise than AVM adaptation; it remains relatively unaffected with increasing levels of noise. There is no significant difference between AVM adapted to clean data and CAT adapted to *BAB05*, *BAB00* or *RVB60*. CAT data adapted to moderate levels of noise can even outperform AVM adapted on clean data, as was seen with CAT *RVB30*.

In contrast, the AVM saw a significant drop in MOS scores with increasing levels of noise, to the extent that AVM *BAB00* is no better than speaker-dependent models trained on the noisy 100 sentences only (SD *BAB00*).

AVM adaptation with noisy data caused the resulting synthesised samples to slow down considerably. For example, the speech of samples synthesised with AVM *BAB00* were on average 51% longer than that synthesised with AVM *CLEAN*. This was due to silent frames for noisy data looking more like speech and thus being consumed by speech models instead of silence or pause models. For example, the proportion of frames assigned to speech (as opposed to silence/pause) during adaptation was 4.8% (relative) higher for *BAB00* than for *CLEAN*.

This resulted in more frames being assigned to speech at synthesis time. As shown in Table 1, different phone types were affected with varying degrees[2]. Voiced obstruents were affected the most (on average 2.6 times longer in *BAB00* than in *CLEAN*). The table shows that silence is the only category for which more frames were allocated to it in the clean condition than in the noisy conditions.

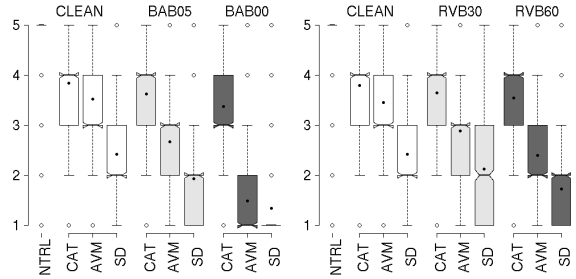
To eliminate any effect on the MOS scores due to the slowing down of the speech, the AVM samples were synthesised using durations obtained from the CAT model and subjective tests re-run. The same overall tendency may be observed in Figure 3: the AVM degrades faster than CAT with increasing noise levels; CAT outperforms the AVM in the presence of noise.

A more constrained form of AVM adaptation with global CMLLR transforms led to less noisy output, compared to AVM with regression class transforms. However, they were still much

phone type	AVM *BAB05*	AVM *BAB00*
voiced obstruents	1.56	2.60
voiceless obstruents	1.21	1.61
nasals	1.37	1.64
other consonants	1.18	1.41
vowels	1.08	1.13
pause	1.26	1.70
silence	0.95	0.95

Table 1: Ratio of average duration per phone for *BAB05*/*CLEAN* and *BAB00*/*CLEAN*, for samples synthesised from the AVM-adapted models, analysed by phone type.

Figure 2: Distribution of mean MOS scores. Black points represent the mean of each distribution.

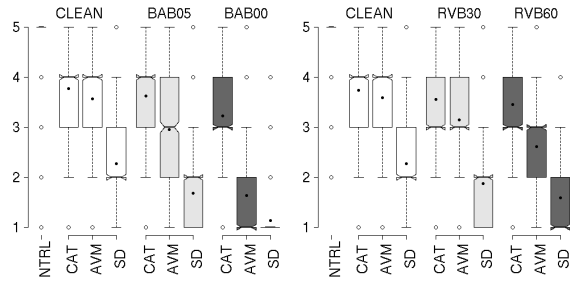


(a) Babble (b) Reverberation

[1]This was based on subjective judgements made by the authors.

[2]Other noises may affect the duration of each phone type differently.

Figure 3: Distribution of mean MOS scores; AVM samples synthesised with CAT duration.



(a) Babble (b) Reverberation

noisier than CAT output. In addition, using global transforms led to more artefacts and decreased similarity.

4. Analysis of speaker similarity in CAT

When building a voice for a target speaker, it is important to assess how similar the synthetic voice is to the original. Under clean conditions, CAT and AVM have been found to perform similarly in terms of speaker similarity [15]. With noisy data for CAT, it was observed informally[1] that speaker similarity degrades with increasing levels of noise, even though the speech quality remained relatively unaffected. The noisier the adaptation data, the more similar the output of different speakers.

It was hypothesised that only a small amount of signal was available for estimating the point in space for the target speaker, due to noise occupying a high proportion of the signal. Thus the speaker subspace was smaller for high levels of noise. This hypothesis may be tested objectively without recourse to subjective tests. In this section, a log-likelihood analysis of the adapted models is performed and multidimensional scaling (MDS) is used to visualise the data.

4.1. Log-likelihood variance analysis

CAT weights for each target speaker were used to align clean data from every other speaker and thus the pairwise log-likelihood of alignment was obtained for each speaker pair. The variance of log-likelihoods was then obtained for each condition as shown in Table 2. It shows that there is less variance in the babble noise conditions than for the clean condition, indicating a smaller speaker subspace. In the reverberation condition, the levels of reverb used in our experiments did not affect the size of the overall subspace as much.

Interestingly, even in noisy conditions, the bimodality of log-likelihoods is retained between male and female speakers. This may be due to log-f0 being relatively unaffected by noise. To test this hypothesis, CAT weights for log-f0 from male speakers were transplanted to female speakers' CAT weights (and vice-versa) and these were used to align data from all the test speakers, in a manner similar to above. This was done for all combinations of speakers.

Analysis was performed by comparing two cases as follows: a) the gender of the data matches the gender of spectral weights but log-f0 weights are of the opposite gender and b) the gender of the data matches the gender of the log-f0 weights but the spectral weights are of the opposite gender. For *CLEAN* the log-likelihood for a) was higher. However, for *BAB00*, b) was marginally higher, confirming our hypothesis that log-f0 plays a

condition	*CLEAN*	*BAB05*	*BAB00*
σ^2 (all)	32.48	25.49	12.21
σ^2 (male)	3.76	3.71	1.81
σ^2 (female)	3.46	2.66	2.09
condition	*CLEAN*	*RVB30*	*RVB60*
σ^2 (all)	27.84	26.67	24.31
σ^2 (male)	4.05	4.19	3.94
σ^2 (female)	3.78	3.85	4.71

Table 2: Variance of each log-likelihood matrix for CAT adaptation. Note that the variances of the *CLEAN* conditions are different because different pre-processing is applied (Section 3.1.2).

big role in determining whether the data aligns better with male or female weights in noisy conditions, as it remains relatively unaffected by noise.

4.2. MDS analysis

Another way to investigate the distribution of voices is to visualise them in a low dimensional space derived from the synthesised speech parameters, using an MDS technique [18]. The axes of the space output by MDS do not have any pre-defined meaning, but MDS attempts to preserve the pairwise distances between the speakers, thus placing similar-sounding speakers close to each other in the space.

Parameters were generated from the adapted CAT model, then mean Mel-LSP distances were calculated for each speaker pair. In order to maintain the same number of frames across all speakers, the generated parameters were constrained using the initial duration CAT weights for all speakers. A distance matrix was created for each noise condition and MDS analyses were performed. Figure 4 shows how the speaker space, indicated by the convex hull, shrinks with increasing levels of noise.

5. Conclusion

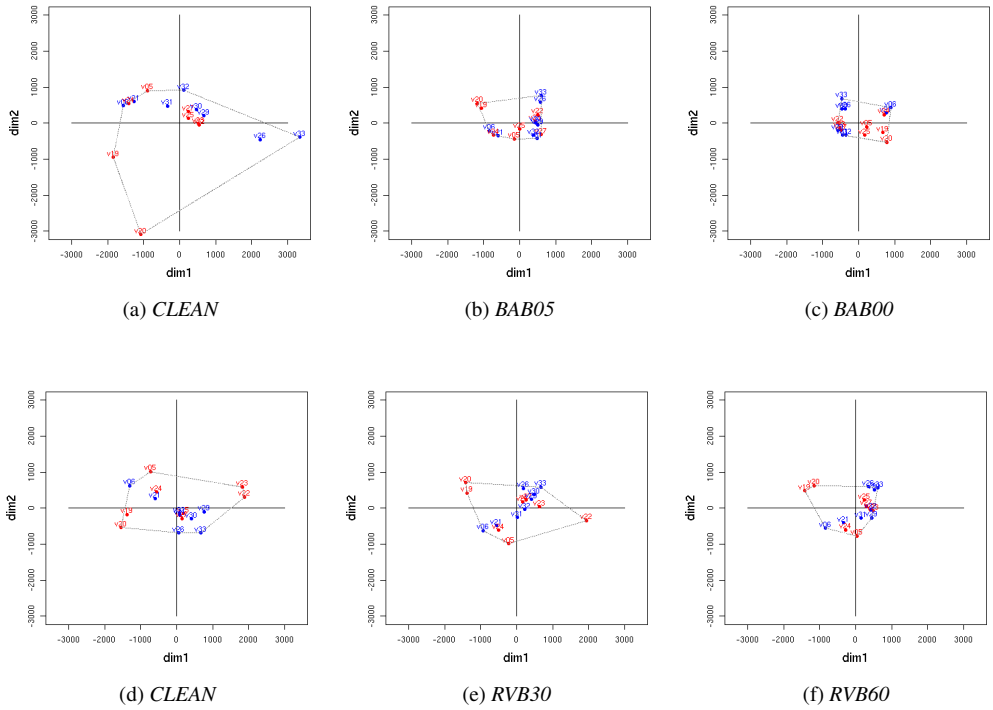
This paper studies the robustness of AVM and CAT adaptation to noisy data. The results of subjective experiments show that AVM suffers significant levels of degradation with noisy adaptation data corrupted with additive noise (babble) and convolutional noise (reverberation). In contrast, in terms of speech quality, CAT is relatively robust to adaptation data with these kinds of corruption.

Pre-processing with spectral subtraction only helps for additive noise and even then there is a limit to how much it can help. While the results would depend on the type and degree of signal processing applied, this indicates that a noise-robust approach to adaptation is still required.

The results confirm the hypothesis that linear transforms used to adapt AVMs are too powerful because noise in the adaptation data is modelled and synthesised in the output speech. The CAT space, on the other hand, is much more constrained so that there is not enough flexibility in the model to deviate much from clean speech, even when the adaptation data is noisy.

This study investigated robustness from the perspective of output speech quality but not speaker similarity. It is known that AVM outperforms CAT in terms of speaker similarity when a large amount of adaptation data is available [15]. With CAT, the speaker subspace became smaller with noisy adaptation data, indicating that speaker similarity is compromised by noise. Subjective evaluation of speaker similarity is left for fu-

Figure 4: MDS visualisation of CAT speaker space computed from Mel-LSP distances between each speaker. Red points represent female speakers and blue points male. See note in Table 2 about pre-processing.



(a) *CLEAN* (b) *BAB05* (c) *BAB00*

(d) *CLEAN* (e) *RVB30* (f) *RVB60*

ture work. Future research will also include the evaluation of a noise-robust approach to produce the context feature labels and to investigate noise factorisation.

6. References

[1] J. Yamagishi, T. Kobayashi, Y. Nakano, K. Ogata, and J. Isogai, "Analysis of Speaker Adaptation Algorithms for HMM-Based Speech Synthesis and a Constrained SMAPLR Adaptation Algorithm," *IEEE Trans. Audio Speech Lang. Process.*, vol. 17, no. 6, pp. 66–83, 2009.

[2] J. Yamagishi, T. Nose, H. Zen, Z.-H. Ling, T. Toda, K. Tokuda, S. King, and S. Renals, "Robust speaker-adaptive HMM-based text-to-speech synthesis," *IEEE Trans. Audio Speech Lang. Process.*, vol. 17, no. 6, pp. 1208–1230, 2009.

[3] J. Yamagishi, M. Lincoln, S. King, J. Dines, M. Gibson, J. Tian, and Y. Guan, "Analysis of unsupervised and noise-robust speaker-adaptive HMM-based speech synthesis systems toward a unified ASR and TTS framework," in *Proc. Blizzard Challenge Workshop*, 2009.

[4] R. Karhila, U. Remes, and M. Kurimo, "HMM-based speech synthesis adaptation using noisy data: analysis and evaluation methods," in *Proc. ICASSP*, 2013.

[5] H. Zen, N. Braunschweiler, S. Buchholz, M. Gales, K. Knill, S. Krstulović, and J. Latorre, "Statistical Parametric Speech Synthesis Based on Speaker and Language Factorization," *IEEE Trans. Audio Speech Lang. Process.*, vol. 20, no. 5, pp. 1713–1724.

[6] T. Nose, Y. Kato, and T. Kobayashi, "A speaker adaptation technique for MRHSMM-based style control of synthetic speech," in *Proc. ICASSP*, 2007, pp. 833–836.

[7] K. Shichiri, A. Sawabe, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, "Eigenvoices for HMM-based speech synthesis," in *Proc. ICSLP*, 2002, pp. 1269–1272.

[8] Y. Ephraim and H. L. V. Trees, "A signal subspace approach for speech enhancement," *IEEE Trans. Speech Audio Process.*, vol. 3, no. 4, pp. 251–266, 1995.

[9] M. Gales, "Cluster adaptive training of hidden Markov models," *IEEE Trans. Speech Audio Process.*, vol. 8, no. 4, pp. 417–428, 2000.

[10] K. Tokuda, T. Masuko, N. Miyazaki, and T. Kobayashi, "Hidden Markov models based on multi-space probability distribution for pitch pattern modeling," in *Proc. ICASSP*, 1999, pp. 229–232.

[11] H. Zen and N. Braunschweiler, "Context-dependent additive log F_0 model for HMM-based speech synthesis," in *Proc. Interspeech*, 2009, pp. 2091–2094.

[12] K. Saino, "A clustering technique for factor analyzed voice models," Master thesis, Nagoya Institute of Technology, 2008.

[13] J. Odell, "The use of context in large vocabulary speech recognition," Ph.D. dissertation, Cambridge University, 1995.

[14] K. Shinoda and T. Watanabe, "Acoustic modeling based on the MDL criterion for speech recognition," in *Proc. Eurospeech*, 1997, pp. 99–102.

[15] V. Wan, J. Latorre, K. K. Chin, L. Chen, M. J. F. Gales, H. Zen, K. Knill, and M. Akamine, "Combining multiple high quality corpora for improving HMM-TTS," in *Proc. Interspeech*, 2012, pp. 1135–1138.

[16] "http://sox.sourceforge.net/."

[17] S. Buchholz, J. Latorre, and K. Yanagisawa, "Crowdsourced assessment of speech synthesis," in *Crowdsourcing for Speech Processing*, M. Eskenazi, G.-A. Levow, H. M. Meng, G. Parent, and D. Suendermann, Eds. Chichester: John Wiley & Sons, 2013, pp. 173–216.

[18] J. Yamagishi, O. Watts, S. King, and B. Usabaev, "Roles of the average voice in speaker-adaptive HMM-based speech synthesis," in *Proc. Interspeech*, 2010, pp. 418–421.

New Method for Rapid Vocal Tract Length Adaptation in HMM-based Speech Synthesis

Daniel Erro [1,2], *Agustín Alonso*[1], *Luis Serrano*[1], *Eva Navas*[1], *Inma Hernáez*[1]

[1] AHOLAB, University of the Basque Country (UPV/EHU), Bilbao, Spain
[2] IKERBASQUE, Basque Foundation for Science, Bilbao, Spain
derro@aholab.ehu.es

Abstract

We present a new method to rapidly adapt the models of a statistical synthesizer to the voice of a new speaker. We apply a relatively simple linear transform that consists of a vocal tract length normalization (VTLN) part and a long-term average cepstral correction part. Despite the logical limitations of this approach, we will show that it effectively reduces the gap between source and target voices with only one reference utterance and without phonetic segmentation. In addition, by using a minimum generation error criterion we avoid some of the problems that have been reported to arise when using a maximum likelihood criterion in VTLN.

Index Terms: statistical parametric speech synthesis, speaker adaptation, vocal tract length normalization

1. Introduction

The length of the vocal tract is one of the perceptually relevant characteristics of a speaker's voice, being known to correlate well with gender and/or age. Therefore, VTLN techniques are useful to make speech processing systems able to operate with a wide variety of voices. VTLN has been traditionally applied to compensate for vocal tract length mismatches between the pre-trained statistical models and the input voices in automatic speech recognition (ASR) [1]. Thus, the word error rate is reduced by 7-10% with respect to an equivalent nonadaptive ASR system.

From the speech generation side, VTLN has also been applied in voice conversion [2] and more recently in speaker-adaptive synthesis [3] to mimic the characteristics of specific target speakers. Frequency warping functions are used to transfer the vocal tract length of the target speaker to the generated speech by modifying either the signal (conversion) or the generative models (synthesis). In this context, VTLN has two main advantages with respect to other types of transformation: (i) the almost null degradation of the quality; (ii) the robustness of the method when few training data are available, which is due to the generally low dimension of the transformation function. These two advantages are often sufficient to justify the use of VTLN in speech generation even though the similarity between frequency-warped and target voices is obviously moderate.

In the particular case of speech synthesis based on hidden Markov models (HMMs), an extensive study was presented in [3] in which the main challenges arising when integrating VTLN in this framework were analyzed. Choosing the popular all-pass transform based on a bilinear function as basic frequency warping curve with only one parameter [4], several model adaptation strategies based on maximum likelihood (ML) criteria were examined. We would like to highlight some observations from the work presented in [3]: (a) the high dimension of the Mel-cepstral vectors typically used in synthesis hinders the adaptation process driven by likelihoods; (b) special attention has to be paid to Jacobian normalization during adaptation to avoid unstabilities; (c) during adaptation, a numerical algorithm is necessary to search for the maximum of an auxiliary function at each iteration of the expectation-maximization algorithm, which results in a doubly iterative procedure.

In our previous works on voice conversion, we showed the usefulness of the so called BLFW+AS (bilinear frequency warping plus amplitude scaling) method [5]. Fed with Mel-cepstral vectors, this method uses a GMM to partition the acoustic vector space of the source speaker into overlapping classes, each class being assigned specific frequency warping and amplitude scaling functions. Like the aforementioned VTLN-based speaker adaptation method, BLFW+AS uses bilinear frequency warping functions with one single parameter. It also uses additive cepstral terms as amplitude scaling functions that compensate for the differences between frequency-warped and target spectra. This paper reports the preliminary steps towards the design of a rapid speaker adaptation method inspired by BLFW+AS in the context of statistical parametric speech synthesis.

Interestingly, although BLFW+AS was designed to operate with multiple overlapping classes, the objective scores presented in [5] (and also those in [3]) suggested that a single frequency warping function followed by many class-dependent amplitude scaling terms performed almost equally well. Therefore, in order to facilitate the design of a BLFW+AS-based adaptation method, in this preliminary approach we have considered only the basic case with a unique transformation class, which means using the same transform in all the acoustic and phonetic contexts. This simplified method can be seen as VTLN followed by a sort of long-term average cepstral correction. In this document, emphasis will be placed on the estimation of the VTLN factor. Future extensions of this work will consider the use of many class-dependent amplitude scaling additive cepstral terms.

Regarding the estimation of the VTLN factor, the method we propose exhibits some remarkable differences with respect to the one described in [3]: (a) our method can deal with high-dimensional Mel-cesptral vectors because it is not based on ML criteria but on a different criterion similar to minimum generation error (MGE) [6]; (b) for the same reason, the Jacobian normalization related problem reported in [3] is avoided; (c) our solution is based on the iterative VTLN training algorithm presented in [7], which converges very rapidly and consistently with no irregular behaviors.

The remainder of this paper is structured as follows. Section 2 summarizes the algorithm that allows calculating a VTLN factor from two parallel sets of cepstral vectors. Then, for a better understanding of the MGE-based method we propose, section 3 will give a brief overview of standard parameter generation techniques used in HMM-based speech

synthesis. In sections 4 and 5 we will describe and evaluate the proposed method assuming one single recorded utterance for adaptation and its corresponding text.

2. Iterative estimation of VTLN factor from aligned vectors

All-pass transforms based on bilinear functions are one of the most popular choices in VTLN [8][4]. This section gives an introduction to this particular type of frequency warping function and shows how the optimal value of its unique parameter can be learnt from a set of aligned source and target vectors, $\{\mathbf{x}_t\}$ and $\{\mathbf{y}_t\}$. Bilinear functions can be defined in the z domain in terms of one single parameter α:

$$z^{(\alpha)^{-1}} = \frac{z^{-1}-\alpha}{1-\alpha z^{-1}}\ ,\quad z = e^{j\omega}\ ,\quad z^{(\alpha)} = e^{j\omega^{(\alpha)}}\ ,\quad |\alpha|<1 \tag{1}$$

The corresponding mapping between the original frequency scale and the warped one is given by

$$\omega^{(\alpha)} = \tan^{-1}\frac{\left(1-\alpha^2\right)\sin\omega}{\left(1+\alpha^2\right)\cos\omega - 2\alpha} \tag{2}$$

Previous research has shown that, given the VTLN factor α, the cepstral representation of a spectrum, $\mathbf{x}$, can be transformed into that of the corresponding frequency-warped spectrum, $\mathbf{x}^{(\alpha)}$. This cepstral transformation can be expressed as a linear operation [4][9]:

$$\mathbf{x}^{(\alpha)} = \mathbf{A}_\alpha \mathbf{x}\ ,\quad \mathbf{A}_\alpha = \begin{bmatrix} 1 & \alpha & \alpha^2 & \cdots \\ 0 & 1-\alpha^2 & 2\alpha-2\alpha^3 & \cdots \\ 0 & -\alpha+\alpha^3 & 1-4\alpha^2+3\alpha^4 & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} \tag{3}$$

Given the strongly nonlinear dependence between $\mathbf{A}_\alpha$ and α, it is difficult to estimate the best value of α from aligned source and target training data. However, since $|\alpha| << 1$ when VTLN is performed on realistic human voices (in general, $|\alpha| < 0.1$), one can think of simplifying $\mathbf{A}_\alpha$ by neglecting the terms of the form α^n for $n > 1$, as originally proposed in [10]. This results in a more manageable transformation:

$$\mathbf{x}^{(\alpha)} \cong \mathbf{x} + \alpha \cdot \mathbf{d}(\mathbf{x}) \tag{4}$$

where $\mathbf{d}(\mathbf{x})$ is the vector whose i^{th} element is given by

$$\mathbf{d}(\mathbf{x})[i] = (i+1)\cdot\mathbf{x}[i+1] - (i-1)\cdot\mathbf{x}[i-1]\ ,\quad i = 0, 1, 2... \tag{5}$$

Under the assumption that (4) is accurate enough, given a set of T source and target parallel training vectors, $\{\mathbf{x}_t\}$ and $\{\mathbf{y}_t\}$, it can be shown [7] that the VTLN factor than minimizes the error between warped and target vectors is

$$\alpha = \frac{\sum_{t=1}^{T}\mathbf{d}^{\mathsf{T}}(\mathbf{x}_t)\cdot(\mathbf{y}_t - \mathbf{x}_t)}{\sum_{t=1}^{T}\left\|\mathbf{d}(\mathbf{x}_t)\right\|^2} \tag{6}$$

In our previous works [7][5] we found expression (4) to be inaccurate for many voices, especially in cross-gender transformation (which is the case where accurate VTLN is most neeeded). Therefore, we proposed the following iterative algorithm to get the minimum-error value of α according to the full formulation (3):

- Step 1: initialize α as 0.
- Step 2: for the current α, calculate a set of warped vectors $\{\mathbf{x}_n^{(\alpha)}\}$, $\mathbf{x}_n^{(\alpha)} = \mathbf{A}_\alpha\mathbf{x}_n$, where the warping matrix $\mathbf{A}_\alpha$ is given by expression (3).
- Step 3: calculate the incremental warping factor $\Delta\alpha$ that is necessary to make the vectors $\{\mathbf{x}_n^{(\alpha)}\}$ closer to the target vectors $\{\mathbf{y}_n\}$. This is done by solving the approximate expression (6) for $\{\mathbf{x}_n^{(\alpha)}\}$ instead of $\{\mathbf{x}_n\}$.
- Step 4: accumulate $\Delta\alpha$ into the current α. This can be done via the following expression [8]:

$$\alpha^{(updated)} = \frac{\alpha + \Delta\alpha}{1 + \alpha\cdot\Delta\alpha} \tag{7}$$

- Step 5: if the last update of α was insignificant (in other words, if $|\Delta\alpha|$ was lower than a threshold), exit. Otherwise, go back to step 2.

3. ML parameter generation from HMMs

For a better understanding of the method to be proposed next, this section explains briefly how the speech parameter generation algorithm of a standard statistical synthesizer [11][12] works. Although usually referred to as HMM-based synthesis, statistical parametric speech synthesis is actually based on context dependent hidden semi Markov models (CD-HSMMs), where the duration of each state is explicitly modeled through normal distributions instead of depending on state transition probabilities. During training, CD-HSMMs are used to model the correspondence between the phonetic, linguistic and prosodic context labels and the observed acoustic parameters (together with their 1st and 2nd-order derivatives over time). During synthesis, once the context labels are extracted from the input text, the system's engine determines the sequence of CD-HSMM states that corresponds to that text and also the duration of each state (either using statistics or specifications by the user). Let us refer to the state index at frame t as m_t. The goal is finding the most probable sequence of acoustic vectors $\{\mathbf{y}_t\}_{t=1\ldots T}$ given the sequence of mean vectors $\{\boldsymbol{\mu}_{m_t}\}_{t=1\ldots T}$ and covariance matrices $\{\boldsymbol{\Sigma}_{m_t}\}_{t=1\ldots T}$. To make the problem mathematically tractable, the output sequence is expressed as a supervector:

$$\mathbf{y} = \left[\mathbf{y}_1^{\mathsf{T}}\quad \mathbf{y}_2^{\mathsf{T}}\quad \ldots\quad \mathbf{y}_T^{\mathsf{T}}\right]^{\mathsf{T}} \tag{8}$$

Defining $\mathbf{W}$ as the matrix that appends dynamic features to the vectors in $\mathbf{y}$ and omitting the derivation (interested readers should refer to [11][12] for details), the most probable $\mathbf{y}$ is

$$\mathbf{y} = \left(\mathbf{W}^{\mathsf{T}}\mathbf{D}\mathbf{W}\right)^{-1}\mathbf{W}^{\mathsf{T}}\mathbf{D}\boldsymbol{\mu} \tag{9}$$

where

$$\boldsymbol{\mu} = \left[\boldsymbol{\mu}_{m_1}^{\mathsf{T}}\quad \boldsymbol{\mu}_{m_2}^{\mathsf{T}}\quad \ldots\quad \boldsymbol{\mu}_{m_T}^{\mathsf{T}}\right]^{\mathsf{T}} \tag{10}$$

and $\mathbf{D}$ is a block-diagonal matrix given by

$$\mathbf{D} = \text{diag}\left\{\boldsymbol{\Sigma}_{m_1}^{-1}, \boldsymbol{\Sigma}_{m_2}^{-1}, \ldots, \boldsymbol{\Sigma}_{m_T}^{-1}\right\} \tag{11}$$

Although the synthesis engine of modern synthesizers includes a global variance enhancement algorithm [13], this is not crucial for our MGE-based adaptation method to perform correctly.

4. Adaptation based on MGE criterion

The algorithms shown in the previous sections provide the necessary tools to build a MGE-based VTLN method in the context of statistical parametric speech synthesis. The idea is (i) to generate a synthetic copy of the utterances available for adaptation and then (ii) to calculate the VTLN factor α that produces the lowest error between warped synthetic utterances and adaptation utterances. For simplicity, we will assume a single adaptation utterance given by the acoustic vector set $\{\mathbf{x}_t\}_{t=1\ldots T}$ and its corresponding text, from which the synthesis engine can determine the sequence of CD-HSMM states to be used. For clarity, we will assign an index to each state in order of appearance: {1, 2, … , M}. Since the iterative algorithm in section 3 requires a set of aligned vectors as input, the state durations must match those of the target utterance. For a more versatile adaptation, it is interesting to perform time-alignment automatically even when a segmentation of that reference utterance is not available. Therefore, in this work we use the synthesis models to obtain such segmentation via forced alignment. A Viterbi search is carried out to establish the correspondence between frames {1…T} and states {1…M} by determining the sequence {m_1 … m_T} that fulfils the continuity and left-to-right conditions ($m_1 = 1$; $m_T = M$; $m_{t+1} = m_t$ or $m_t + 1$ for all t) and maximizes the following log-likelihood function:

$$C\{m_1, m_2 \ldots m_T\} = \sum_{t=1}^{T} \log N(\mathbf{X}_t; \boldsymbol{\mu}_{m_t}, \boldsymbol{\Sigma}_{m_t}) + \\ + \sum_{t=1}^{T-1} \log p(m_{t+1}/m_t, d_t) \quad (12)$$

where $\mathbf{X}_t$ is the result of appending dynamic features to $\mathbf{x}_t$, d_t is the duration of state m_t until frame t (it can be obtained recursively: $d_t = 1$ if $t = 1$ or $m_t \neq m_{t-1}$; $d_t = d_{t-1} + 1$ elsewhere), N denotes the normal distribution, and

$$p(m_{t+1}/m_t, d_t) = \begin{cases} \theta_{m_t, d_t} & , \; m_{t+1} = m_t \\ 1 - \theta_{m_t, d_t} & , \; m_{t+1} \neq m_t \end{cases} \quad (13)$$

$$\theta_{m_t, d_t} = \int_{d_t}^{\infty} N(\lambda; \mu_{m_t}^{(d)}, \sigma_{m_t}^{(d)2}) d\lambda$$

Note that (13) means calculating the probability that the duration of the current state is greater than it was at frame t according to the duration means $\mu^{(d)}$ and variances $\sigma^{(d)2}$ learnt during CD-HSMMs training.

Once the state durations are known, ML parameter generation (9)–(11) can be applied to obtain a synthetic version of the adaptation material, $\{\mathbf{y}_t\}_{t=1\ldots T}$, $\{\mathbf{x}_t\}$ and $\{\mathbf{y}_t\}$ being completely parallel. In these conditions, it would be straightforward to obtain the necessary VTLN factor α by means of the algorithm in section 2. The main problem of this approach is that high vocal tract length contrasts between the source (synthetic) voice and the target voice may result in inaccurate state durations and therefore inaccurate α. To avoid it, the following iterative algorithm is applied to jointly optimize α and the durations:

- Step 1: initialize α as 0.
- Step 2: for the current α, calculate a set of frequency-warped adaptation vectors $\{\mathbf{x}_t^{(\alpha)}\}$, $\mathbf{x}_t^{(\alpha)} = \mathbf{A}_\alpha \mathbf{x}_t$, where the involved matrix is given by expression (3).
- Step 3: use the forced alignment method described above to determine the state durations using $\{\mathbf{x}_t^{(\alpha)}\}$ (not $\{\mathbf{x}_t\}$) as reference; then generate $\{\mathbf{y}_t\}$ through expressions (9)–(11).
- Step 4: calculate a new α using the iterative method in section 2, taking the adaptation vectors $\{\mathbf{x}_t\}$ as source and the synthetic vectors $\{\mathbf{y}_t\}$ as target (although this is the opposite direction to the desired one, it simplifies the calculations substantially). Note that the current $\{\mathbf{y}_t\}$ depends on the current durations, which in turn depend on the current α (we avoid more specific notation for clarity).
- Step 5: if the last update of α was insignificant, multiply α by -1 (this means inverting the warping function, thus making it suitable to transform the synthetic voice into the target voice) and exit. Otherwise, go back to step 2.

Under the assumption that the state durations and the resulting set $\{\mathbf{y}_t\}$ have converged together with the VTLN factor α, an additive cepstral correction term is calculated as a complement for VTLN:

$$\mathbf{b} = \frac{1}{T} \sum_{t=1}^{T} (\mathbf{x}_t - \mathbf{A}_\alpha \mathbf{y}_t) \quad (14)$$

The use of this additive term can be seen as as sort of long-term average spectrum normalization. The final adaptation of the mean vectors $\{\boldsymbol{\mu}_m\}$ and covariance matrices $\{\boldsymbol{\Sigma}_m\}$ at every state of the trained CD-HSMMs is carried out as follows:

$$\hat{\boldsymbol{\mu}}_m = \left[(\mathbf{A}_\alpha \boldsymbol{\mu}_m^{(s)} + \mathbf{b})^\intercal \quad (\mathbf{A}_\alpha \boldsymbol{\mu}_m^{(\Delta)})^\intercal \quad (\mathbf{A}_\alpha \boldsymbol{\mu}_m^{(\Delta\Delta)})^\intercal \right]^\intercal \\ \hat{\boldsymbol{\Sigma}}_m = \mathrm{diag}\{\mathbf{A}_\alpha \boldsymbol{\Sigma}_m^{(s)} \mathbf{A}_\alpha^\intercal, \; \mathbf{A}_\alpha \boldsymbol{\Sigma}_m^{(\Delta)} \mathbf{A}_\alpha^\intercal, \; \mathbf{A}_\alpha \boldsymbol{\Sigma}_m^{(\Delta\Delta)} \mathbf{A}_\alpha^\intercal\} \quad (15)$$

where diag{…} denotes a block-diagonal matrix and (s), (Δ) and (ΔΔ) denote the sub-parts of the vectors/matrices related to static features, their 1st derivatives and their 2nd derivatives, respectively.

Interestingly, we found the described method to perform better when the involved vectors are weighted by the local probability of voicing when calculating both α and $\mathbf{b}$ through expressions (6) and (14), respectively. This prevents long silences and unvoiced segments from biasing the results too much. In HMM-based speech synthesis, the probability of voicing at each state can be easily extracted from the weights of the multi-space distributions (MSD) used to model/generate the $\log f_0$ contour [14].

Finally, an average pitch modification factor is calculated by generating a synthetic $\log f_0$ contour according to the last instance of the state durations and comparing it with that of the adaptation utterance. In this case, adaptation is performed by summing the appropriate constant value to the static part of the mean vectors of the CD-MSD-HSMMs trained from $\log f_0$.

5. Preliminary evaluation

As discussed in [3], evaluating VTLN is not an easy task because the behavior of the method depends on the specific voices involved in the test. In addition, performing VTLN followed by long-term average cepstral correction implies modifying just a few characteristics of the source voice. Therefore, for arbitrary input voices and large amounts of data the proposed adaptation method cannot compete with more sophisticated methods such as the well known CSMAPLR [15]. On the other hand, we would also like to emphasize that these are just the preliminary steps towards the design of a better method inspired by the BLFW+AS voice conversion

method [5]. Taking all this into account, we have conducted a relatively simple perceptual test to show that (i) our method can effectively reduce the gap between the voice of a synthesizer and a given target voice and (ii) it can do it rapidly using only one reference utterance and its corresponding text.

The text-to-speech (TTS) synthesis system used in our experiments, AhoTTS [16], includes a statistical engine based on HTS [17] and a Mel-cepstral vocoder based on a harmonics plus noise model [18]. The default voice of the system was trained from 2k utterances recorded from a female speaker in Castilian Spanish and digitized at 16 kHz sampling frequency. In previous informal listening tests we had found this voice to be quite suitable for VTLN-based transformations, even towards male voices. We recorded one short utterance (9 words) from 11 different non-professional speakers (5 female plus 6 male speakers). Using them as target, we applied our adaptation method to transform the models of AhoTTS's default voice and then we synthesized speech in all of these voices. Next, 15 volunteer listeners (half of them were speech processing experts) rated the following aspects on a 5-point scale: similarity between the default synthetic voice and the target natural voice, similarity between the adapted synthetic voice and the target natural voice, and relative quality of the adapted voice with respect to the default synthetic voice. As usual, the score indicating the lowest similarity/quality is 1 and the highest score is 5.

Table 1. *Results of the perceptual test: MOS and 95% confidence interval.*

	Source-target Sim.	Adapted-target Sim.	Quality of adapted
Intra-gender	1.40 ± 0.16	2.78 ± 0.25	4.12 ± 0.25
Cross-gender	1.03 ± 0.04	2.63 ± 0.20	3.74 ± 0.22
Total average	1.20 ± 0.08	2.69 ± 0.16	3.94 ± 0.17

The mean opinion scores (MOSs) summarized in Table 1 indicate that, despite the evident differences between source and target voices (~1.2 similarity MOS on a 1-to-5 scale) and the low amount of training material, the proposed method makes the adapted voice significantly closer to the target (~2.7 similarity MOS). The quality loss due to the adaptation process is not particularly high (~4 relative quality MOS). Given the differences between intra-gender and cross-gender cases, we believe that this apparent 1-point quality gap is partially related to the naturalness of the voice that results from this particular type of adaptation rather than to the appearance of artifacts. Overall, taking into account the nature of the method and despite the absence of baseline methods in the listening test, these MOSs reveal that our research goes in the right direction and also that the method proposed in this preliminary work still needs to be improved in order to achieve more satisfactory similarity MOSs.

6. Conclusions

We have presented a new method to adapt the voice of an HMM-based synthesizer using a single unlabeled utterance from the target speaker and its corresponding text. Despite using a relatively simple transformation consisting of VTLN, long-term average cepstral correction and pitch shifting, the method succeeds at reducing the distance between adapted and target voices significantly. Future works will aim at improving the similarity scores achieved by the method through the use of class-dependent additive cepstral terms.

7. Acknowledgements

This work has been partially supported by the Spanish Ministry of Economy and Competitiveness (SpeechTech4All, TEC2012-38939-C03-03), the Basque Government (Ber2tek, IE12-333) and Euroregion Aquitaine-Euskadi (Iparrahotsa, 2012-004).

8. References

[1] P. Zhan, A. Waibel, "Vocal tract length normalization for large vocabulary continuous speech recognition", CMU Computer Science Tech. Rep., 1997.

[2] D. Sündermann, H. Ney, "VTLN-based voice conversion", Proc. IEEE Symp. Signal Process. Inf. Technol., pp. 556–559, 2003.

[3] L. Saheer, J. Dines, P. Garner, "Vocal Tract Length Normalization for Statistical Parametric Speech Synthesis", IEEE Trans. Audio, Speech and Lang. Process., vol. 20(7), pp. 2134-2148, 2012.

[4] J. McDonough, W. Byrne, "Speaker adaptation with all-pass transforms", Proc. ICASSP, pp. 757–760, 1999.

[5] D. Erro, E. Navas, I. Hernaez, "Parametric Voice Conversion Based on Bilinear Frequency Warping Plus Amplitude Scaling", IEEE Trans. Audio, Speech and Lang. Process., vol. 21(3), pp. 556-566, 2012.

[6] L. Qin, Y.J. Wu, Z.H. Ling, R.H. Wang, L.R. Dai, "Minimum generation error linear regression based model adaptation for HMM-based speech synthesis", Proc. ICASSP, pp. 3953-3956, 2008.

[7] D. Erro, E. Navas, I. Hernaez, "Iterative MMSE Estimation of Vocal Tract Length Normalization Factors for Voice Transformation", Proc. Interspeech, pp. 86-89, 2012.

[8] A. Acero, "Acoustical and environmental robustness for automatic speech recognition", Ph.D. dissertation, Carnegie Mellon Univ., Pittsburgh, PA, 1990.

[9] M. Pitz, H. Ney, "Vocal tract normalization equals linear transformation in cepstral space", IEEE Trans. Speech Audio Process., vol. 13(5), pp. 930–944, 2005.

[10] T. Emori, K. Shinoda, "Rapid vocal tract length normalization using maximum likelihood estimation", Proc. Eurospeech, pp. 1649–1652, 2001.

[11] K. Tokuda, T. Yoshimura, T. Masuko, T. Kobayashi, T. Kitamura, "Speech parameter generation algorithms for HMM-based speech synthesis", Proc. ICASSP, vol. 3, pp. 1315-1318, 2000.

[12] H. Zen, K. Tokuda, A. W. Black, "Statistical parametric speech synthesis", Speech Commun., vol. 51(11), pp. 1039-1064, 2009.

[13] T. Toda, K. Tokuda, "A Speech Parameter Generation Algorithm Considering Global Variance for HMM-Based Speech Synthesis", IEICE Trans. Info. & Syst., vol. E90-D(5), pp. 816-814, 2007.

[14] K. Tokuda, T. Masuko, N. Miyazaki, T. Kobayashi, "Hidden Markov models based on multi-space probability distribution for pitch pattern modeling", Proc. ICASSP, pp. 229-232, 1999.

[15] J. Yamagishi, T. Kobayashi, Y. Nakano, K. Ogata, J. Isogai, "Analysis of speaker adaptation algorithms for HMM-based speech synthesis and a constrained SMAPLR adaptation algorithm," IEEE Trans. Audio, Speech, Lang. Process., vol. 17(1), pp. 66–83, 2009.

[16] I. Sainz, D. Erro, E. Navas, I. Hernaez, J. Sanchez, I. Saratxaga, I. Odriozola, I. Luengo, "Aholab Speech Synthesizers for Albayzin 2010", Proc. FALA, pp. 343-347, 2010.

[17] Online: "HMM-based Speech Synthesis System (HTS)", http://hts.sp.nitech.ac.jp/

[18] D. Erro, I. Sainz, E. Navas, I. Hernaez, "Improved HNM-based vocoder for statistical synthesizers", Proc. InterSpeech, pp. 1809-1812, 2011.

Text-to-speech synthesizer based on combination of composite wavelet and hidden Markov models

Nobukatsu Hojo[1], *Kota Yoshizato*[1], *Hirokazu Kameoka*[1,2], *Daisuke Saito*[1], *Shigeki Sagayama*[1,*]

[1]Graduate School of Information Science and Technology, the University of Tokyo, Japan
[2] Communication Science Laboratories, NTT Corporation, Japan
{hojo, yoshizato, kameoka, dsaito, sagayama}@hil.t.u-tokyo.ac.jp

Abstract

This paper proposes a text-to-speech synthesis (TTS) system based on a combined model consisting of the Composite Wavelet Model (CWM) and the Hidden Markov Model (HMM). Conventional HMM-based TTS systems using cepstral features tend to produce over-smoothed spectra, which often result in muffled and buzzy synthesized speech. This is simply caused by the averaging of spectra associated with each phoneme during the learning process. To avoid the over-smoothing of generated spectra, we consider it important to focus on a different representation of the generative process of speech spectra. In particular, we choose to characterize speech spectra using the CWM, whose parameters correspond to the frequency, gain and peakiness of each underlying formant. This idea is motivated by our expectation that the averaging of these parameters would not lead directly to the over-smoothing of spectra, as opposed to the cepstral representations. To describe the entire generative process of a sequence of speech spectra, we combine the generative process of a formant trajectory using an HMM and the generative process of a speech spectrum using the CWM. A parameter learning algorithm for this combined model is derived based on an auxiliary function approach. We confirmed through experiments that our speech synthesis system was able to generate speech spectra with clear peaks and dips, which resulted in natural-sounding synthetic speech.
Index Terms: text-to-speech synthesis, hidden Markov model, composite wavelet model, formant, Gaussian mixture model, auxiliary function

1. Introduction

This paper proposes a new model for text-to-speech synthesis (TTS). One promising approach for TTS involves methods based on statistical models. In this approach, the first step is to formulate a generative model of a sequence of speech features. The second step is to train the parameters of the assumed generative model given a set of training data in a speech corpus. The third step is to produce the most likely sequence of speech features given a text input and transform it into a speech signal. With this approach, one key to success is that the assumed generative model reflects the nature of real speech well. To model the entire temporal evolution of speech features, a hidden Markov model (HMM) and its variants including the "trajectory HMM" have been introduced with notable success [1, 2, 3]. HMMs are roughly characterized by the structure of the state transition network (i.e., a transition matrix) and an output distribution assumption. In conventional HMM-based TTS systems, a Gaussian mixture emission distribution of a cepstral feature [1, 2] or a line spectrum pair (LSP) feature [4, 3] is typically used as the output distribution of each state. The aim of this paper is to seek an alternative to the conventional speech feature and the state output distribution. Namely, this paper is concerned with formulating a new model for the generative process of speech spectra and combining it with an HMM.

The Gaussian distribution assumption of a cepstral feature describes probabilistic fluctuations of spectra in the power direction, while that of an LSP feature describes probabilistic fluctuations of spectral peaks in the frequency direction. Since the frequencies and powers of peaks in a spectral envelope correspond to the resonance frequencies and gains of the vocal tract, they both vary continuously over time according to the physical movement of the vocal tract. In particular, resonance frequencies and gains both vary significantly in the boundary between one phoneme and another. To achieve higher quality speech synthesis, we consider it important to describe a generative model that takes account of the fluctuations of spectral peaks in both the frequency and power directions rather than a fluctuation in just one direction. Conventional HMM-based TTS systems using cepstral features tend to produce over-smoothed spectra, which often result in muffled and buzzy synthesized speech. This is simply caused by the averaging of observed log-spectra assigned to each state during the training process (Fig. 2 (a)). Although some attempts were made to emphasize the peaks and dips of generated spectra via post-processing [5], it is generally difficult to restore original peaks and dips once spectra are over-smoothed. By contrast, this paper proposes tackling this problem by introducing a different spectral representation called the Composite Wavelet Model (CWM) [6, 7, 8] as an alternative to cepstral and LSP representations, on which basis we formulate a new generative model for TTS.

2. Generative model of spectral sequence

2.1. Motivation

Taking the mean of cepstral features associated with a particular phoneme amounts to taking the mean of the corresponding log spectra. Since the resonance frequencies and gains of the vocal tract both vary continuously during the transition from one phoneme to another, if we simply take the mean of the log spectra, the spectral peaks and dips will be be blurred and indistinct (Fig. 2 (a)). By contrast, if we can appropriately treat the frequencies and powers of individual spectral peaks as speech features, we expect that taking their means will not cause the blurring of spectra (Fig. 2 (b)). CWM approximates a speech spectral envelope from the sum of the Gaussian distributions,

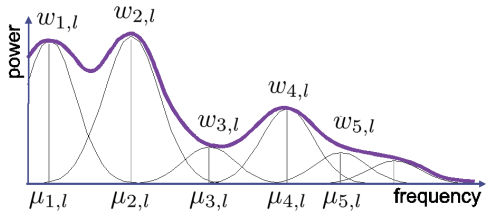
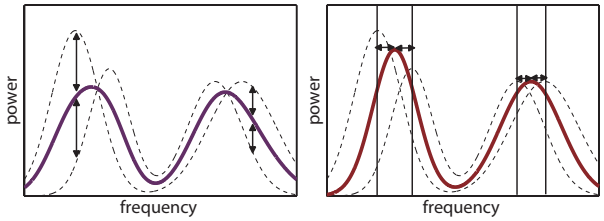


Figure 1: Spectral representation based on CWM



(a) Mean in cepstral domain (b) Mean in CWM domain

Figure 2: Expectations of spectra taken in different domains. The dashed lines indicate the training samples and the solid lines indicate the mean spectra.

interpreted as a function of frequency (see Fig. 1 for a graphical illustration). This means each Gaussian distribution function roughly corresponds to a peak in a spectral envelope. CWM is thus convenient for describing both the frequency and power fluctuations of spectral peaks.

Another notable feature of CWM is that its parameters can be easily transformed into a signal. Since a Gaussian function in the frequency domain corresponds to a Gabor function in the time domain, CWM parameters can be directly converted into a signal by superimposing the corresponding Gabor functions. CWM can thus be regarded as a speech synthesizer with an FIR filter and its superiority to conventional systems with IIR filters has already been shown in [6].

One straightforward way of incorporating CWM into an HMM-based TTS system would be to first perform CWM parameter extraction on a frame-by-frame basis, and then train the parameters of an HMM by treating the extracted CWM parameters as a sequence of feature vectors. However, our preliminary investigation revealed that this simple method did not work well [9]. One reason for this is the common difficulty of formant tracking. Although formants and their time courses (formant trajectories) are clearly visible to the human eye in the spectrum and in the spectrogram, automatic formant extraction and tracking are far from trivial. Many formant extraction algorithms miss a formant that is present, insert a formant when there is none, or mislabel them (such as label F_1 as F_2, or F_3 as F_2), mostly as a result of incorrectly pruning the correct formant at the frame level. This is also the case for frame-by-frame CWM parameter extraction since it can be thought of as a sort of formant extraction. Fig. 3 shows an example of results obtained with frame-by-frame CWM analysis. As this example shows, the index of each Gaussian distribution function in the CWM at one particular frame is not always consistent with that at another frame. This was problematic when training the HMM parameters since the mean of the emission distribution of each state is given as the average of the feature vectors assigned to the same state. To train the HMM parameters appropriately, the indices of the Gaussian functions of the CWM assigned to the same state must always be consistent. This implies the need for the joint modeling of the CWM and HMM. Motivated by the above discussion, we here describe the entire generative process of a sequence of speech spectra by combining the generative process of a speech spectrum based on the CWM with the generative process of a CWM parameter trajectory based on an HMM.

2.2. Formulation

The CWM consists of parameters (roughly) corresponding to the frequency and power of spectral peaks. Specifically, the CWM approximates a spectral envelope by employing a Gaussian mixture model (GMM) interpreted as a function of frequency. Namely, the CWM $f_{\omega,l}$ is defined as

$$f_{\omega,l} = \sum_{k=1}^{K} \frac{w_{k,l}}{\sqrt{2\pi}\sigma_{k,l}} \exp\left(- \frac{(\omega - \mu_{k,l})^2}{2\sigma_{k,l}^2} \right), \tag{1}$$

where ω, l denotes the indices of frequency and time, respectively, and K is the number of mixture components of the GMM. See Fig. 1 for a graphical illustration. $\mu_{k,l}, \sigma_{k,l}$ and $w_{k,l}$ are the mean, variance and weight of each Gaussian (when interpreted as a probability density function), and thus correspond to the frequency, peakiness and power of the peaks in a spectral envelope, respectively.

Using the CWM representation given above as a basis, here we describe the generative process of an entire sequence of observed spectra. We consider an HMM that generates a set consisting of $\mu_{k,l}$, $\rho_{k,l} := 1/\sigma_{k,l}^2$, and $w_{k,l}$ at time l (see Fig. 4). Each state of the HMM represents a label indicating linguistic information, which can be simply either a phoneme label as shown in Fig. 4 or a context label (as used in [10]). Given state s_l at time l, we assume that the CWM parameters are generated according to

$$P(\mu_{k,l}|s_l) = \mathcal{N}(\mu_{k,l}; m_{k,s_l}, \eta_{k,s_l}^2), \tag{2}$$

$$P(\rho_{k,l}|s_l) = \mathrm{Gamma}(\rho_{k,l}; a_{k,s_l}^{(\rho)}, b_{k,s_l}^{(\rho)}), \tag{3}$$

$$P(w_{k,l}|s_l) = \mathrm{Gamma}(w_{k,l}; a_{k,s_l}^{(w)}, b_{k,s_l}^{(w)}), \tag{4}$$

where $\mathcal{N}(x; m, \eta^2)$ denotes the normal distribution and $\mathrm{Gamma}(x; a, b)$ the gamma distribution:

$$\mathrm{Gamma}(x; a, b) = x^{a-1} \frac{\exp(-x/b)}{\Gamma(a)\, b^a}. \tag{5}$$

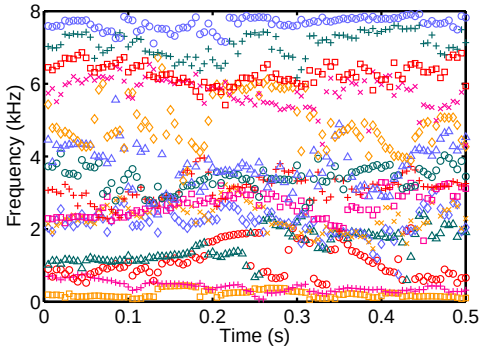
These distribution families are chosen for simplifying the derivation of the parameter estimation algorithm described in the next section. Given the following sequence of CWM parameters, $\boldsymbol{\mu} = \{\mu_{k,l}\}_{k,l}, \boldsymbol{\rho} = \{\rho_{k,l}\}_{k,l}, \boldsymbol{w} = \{w_{k,l}\}_{k,l}$, we further assume that a spectrum $\{y_{\omega,l}\}_\omega$ observed at time l is generated according to

$$P(y_{\omega,l}|\boldsymbol{\mu}, \boldsymbol{\rho}, \boldsymbol{w}) = \mathrm{Poisson}(y_{\omega,l}; f_{\omega,l}), \tag{6}$$

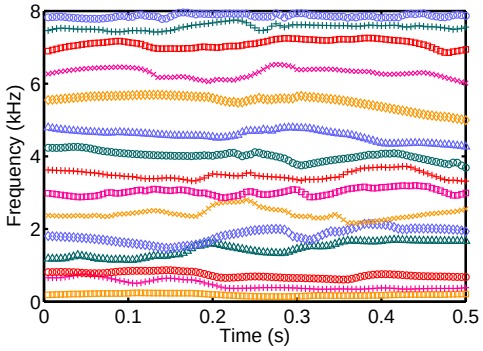
where $\mathrm{Poisson}(x; \lambda)$ denotes the Poisson distribution:

$$\mathrm{Poisson}(x; \lambda) = \frac{\lambda^x e^{-\lambda}}{x!}. \tag{7}$$

This assumption is also made for simplifying the derivation of the parameter estimation algorithm described in the next section. $f_{\omega,l}$ denotes the sequence of spectrum models defined by



(a) The frame-independent CWM parameter extraction method [9]



(b) The proposed method

Figure 3: An example of the formant frequencies extracted using (a) the frame-independent CWM parameter extraction method [9] and (b) the proposed method. The extracted formant frequencies are plotted with different colors according to the indices of the Gaussian distribution functions in CWM.

Eq. 1. It should be noted that the maximization of the Poisson likelihood with respect to $f_{\omega,l}$ amounts to optimally fitting $f_{\omega,l}$ to $y_{\omega,l}$ by using the I-divergence as the fitting criterion [8]. The next section describes the parameter estimation algorithm of the generative model presented above.

3. Parameter estimation algorithm

Here we describe the parameter learning algorithm given a set of training data. The spectral sequence of the training data is considered to be an observed sequence and a set of unknown parameters of the present generative model is denoted by Θ. Θ consists of the state sequence $\boldsymbol{s} = \{s_l\}_l$, the parameters of the state emission distributions $\boldsymbol{\theta} = \{m_{k,i}, \eta_{k,i}, a^{(\rho)}_{k,i}, b^{(\rho)}_{k,i}, a^{(w)}_{k,i}, b^{(w)}_{k,i}\}_{k,i}$, and the CWM parameters $\{\boldsymbol{\mu}, \boldsymbol{\rho}, \boldsymbol{w}\}$.

Given a set of observed spectra $Y = \{y_{\omega,l}\}_{\omega,l}$, we would like to determine the estimate of Θ that maximizes the posterior density $P(\Theta|Y) \propto P(Y|\Theta)P(\Theta)$, or equivalently, $\log P(Y|\Theta) + \log P(\Theta)$. Here, $\log P(\Theta|Y)$ is written as

$$\log P(\Theta|Y) \stackrel{c}{=} \log P(Y|\Theta) + \alpha \log P(\Theta), \tag{8}$$

$$\log P(\Theta) \stackrel{c}{=} \log P(\boldsymbol{s}) + \log P(\boldsymbol{\mu}|\boldsymbol{s}, \boldsymbol{\theta}) + \log P(\boldsymbol{\rho}|\boldsymbol{s}, \boldsymbol{\theta}) + \log P(\boldsymbol{w}|\boldsymbol{s}, \boldsymbol{\theta}), \tag{9}$$

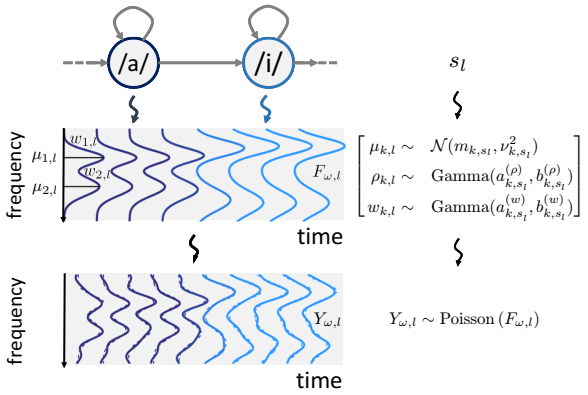


Figure 4: Illustration of the present HMM

where α is a regularization parameter that weighs the importance of the log prior density relative to the log-likelihood. $\stackrel{c}{=}$ denotes equality up to constant terms. Unfortunately, this optimization problem is non-convex, and finding the global optimum is an intractable problem. However, we can employ an auxiliary function approach to find a local optimum [8].

As mentioned above, we notice from Eqs. (6) and (7) that $-\log P(Y|\Theta)$ is equal up to constant terms to the sum of the I-divergence between $y_{\omega,l}$ and $f_{\omega,l}$

$$\mathcal{I}(\Theta) := \sum_{\omega,l} \Big(y_{\omega,l} \log \frac{y_{\omega,l}}{f_{\omega,l}} - y_{\omega,l} + f_{\omega,l} \Big). \tag{10}$$

Hence, maximizing $P(\Theta|Y)$ amounts to minimizing $I(\Theta) - \alpha \log P(\Theta)$ with respect to Θ. By invoking Jensen's inequality based on the concavity of the logarithm function, we obtain an inequality

$$\mathcal{I}(\Theta) = \sum_{\omega,l} \Big(y_{\omega,l} \log \frac{y_{\omega,l}}{f_{\omega,l}} - y_{\omega,l} + f_{\omega,l} \Big) \tag{11}$$

$$\leq \sum_{\omega,l} \Big\{ y_{\omega,l} \log y_{\omega,l} - y_{\omega,l} - \sum_k \Big(\lambda_{k,\omega,l} \log \frac{g_{k,\omega,l}}{\lambda_{k,\omega,l}} + g_{k,\omega,l} \Big) \Big\}, \tag{12}$$

where

$$g_{k,\omega,l} = \sqrt{\frac{\rho_{k,l}}{2\pi}} \exp\Big(-\frac{\rho_{k,l}}{2} (\omega - \mu_{k,l})^2 \Big). \tag{13}$$

By using $\mathcal{J}(\Theta, \lambda)$ to denote the upper bound of $\mathcal{I}(\Theta)$, i.e., the right-hand side of (12), equality holds if and only if

$$\lambda_{k,\omega,l} = \frac{w_{k,l} g_{k,\omega,l}}{\sum_{j=1}^K w_{j,l} g_{j,\omega,l}}. \tag{14}$$

Now, when $\lambda_{k,\omega,l}$ is given by Eq. (14) with an arbitrary Θ, the auxiliary function $\mathcal{J}(\Theta, \lambda) - \alpha \log P(\Theta)$ becomes equal to the objective function $\mathcal{I}(\Theta) - \alpha \log P(\Theta)$. Then, the parameter that decreases $\mathcal{J}(\Theta, \lambda) - \alpha \log P(\Theta)$ while keeping λ fixed will necessarily decrease $\mathcal{I}(\Theta) - \alpha \log P(\Theta)$, since inequation (12) guarantees that the original objective function is always even

smaller than the decreased $\mathcal{J}(\Theta, \lambda) - \alpha \log P(\Theta)$. Therefore, by repeating the update of λ by Eq. (14) and the update of Θ that decreases $\mathcal{J}(\Theta, \lambda) - \alpha \log P(\Theta)$, the objective function decreases monotonically and converges to a stationary point.

With fixed $\boldsymbol{s}$ and $\boldsymbol{\theta}$, the auxiliary function $\mathcal{J}(\Theta, \lambda) - \alpha \log P(\Theta)$ is minimized with respect to the CWM parameters, $\boldsymbol{\mu}$, $\boldsymbol{\rho}$, and $\boldsymbol{w}$, under the following updates

$$\mu_{k,l} = \frac{D_{k,l}\eta_{k,i}^2\rho_{k,l} + \alpha m_{k,i}}{C_{k,l}\eta_{k,i}^2\rho_{k,l} + \alpha}, \tag{15}$$

$$\rho_{k,l} = \frac{C_{k,l} + 2\alpha(a_{k,l}^{(\rho)} - 1)}{C_{k,l}\mu_{k,l}^2 - 2D_{k,l}\mu_{k,l} + E_{k,l} - 2\alpha/b_{k,l}^{(\rho)}}, \tag{16}$$

$$w_{k,l} = \frac{C_{k,l} + \alpha(a_{k,l}^{(w)} - 1)}{1 + \alpha/b_{k,l}^{(w)}}, \tag{17}$$

where

$$C_{k,l} = \sum_{\omega} y_{\omega,l}\lambda_{k,\omega,l}, \tag{18}$$

$$D_{k,l} = \sum_{\omega} \omega y_{\omega,l}\lambda_{k,\omega,l}, \tag{19}$$

$$E_{k,l} = \sum_{\omega} \omega^2 y_{\omega,l}\lambda_{k,\omega,l}. \tag{20}$$

With fixed $\boldsymbol{\mu}$, $\boldsymbol{\rho}$, $\boldsymbol{w}$ and $\boldsymbol{\theta}$, the state sequence minimizing $\mathcal{J}(\Theta, \lambda) - \alpha \log P(\Theta)$ can be obtained by using the Viterbi algorithm. With fixed $\boldsymbol{\mu}$, $\boldsymbol{\rho}$, $\boldsymbol{w}$ and $\boldsymbol{s}$, the auxiliary function is minimized under the following updates.

As for $\{m_{k,i}, \nu_{k,i}^2\}_{k,i}$,

$$m_{k,i} = \frac{1}{N_i} \sum_{l \in \mathcal{T}_i} \mu_{k,l} \tag{21}$$

$$\nu_{k,i}^2 = \frac{1}{N_i} \sum_{l \in \mathcal{T}_i} \mu_{k,l}^2 - (\frac{1}{N_i} \sum_{l \in \mathcal{T}_i} \mu_{k,l})^2, \tag{22}$$

where $\mathcal{T}_i = \{l | s_l = i\}$. As for $\{a_{k,i}^{(\rho)}, b_{k,i}^{(\rho)}, a_{k,i}^{(w)}\}_{k,l}, b_{k,i}^{(w)}\}_{k,l}$, although the updates are not derived in a closed form, they are updated as the root of the following equations

$$\log a_{k,i}^{(\rho)} - \psi(a_{k,i}^{(\rho)}) = \log(\frac{1}{N_i} \sum_{l \in \mathcal{T}_i} \rho_{k,l}) - \frac{1}{N_i} \sum_{l \in \mathcal{T}_i} \rho_{k,l} \tag{23}$$

$$\log a_{k,i}^{(w)} - \psi(a_{k,i}^{(w)}) = \log(\frac{1}{N_i} \sum_{l \in \mathcal{T}_i} w_{k,l}) - \frac{1}{N_i} \sum_{l \in \mathcal{T}_i} w_{k,l} \tag{24}$$

$$b_{k,i}^{(\rho)} = \frac{\frac{1}{N_i} \sum_{l \in \mathcal{T}_i} \rho_{k,l}}{a_{k,i}^{(\rho)}} \tag{25}$$

$$b_{k,i}^{(w)} = \frac{\frac{1}{N_i} \sum_{l \in \mathcal{T}_i} w_{k,l}}{a_{k,i}^{(w)}}, \tag{26}$$

where $\psi(a)$ denotes the digamma function

$$\psi(a) = \frac{\partial \Gamma(a)}{\partial a} / \Gamma(a). \tag{27}$$

A spectrogram of sentence A01 from the ATR503 data set 'Arayurugenjitsuwo subete jibunnohohe nejimagetanoda' and an example of the trajectory of the mean parameters of the CWM extracted from it are shown in Fig. 5.

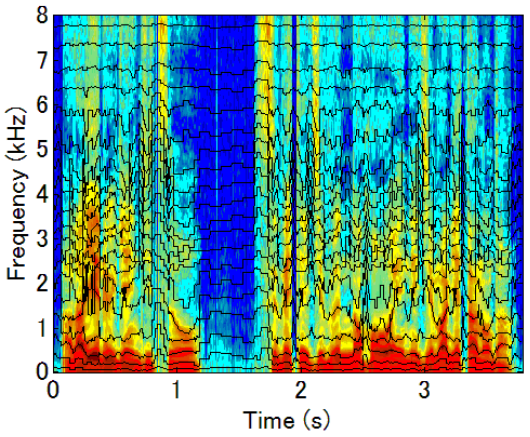


Figure 5: Spectrogram of a natural voice and the trajectories of extracted frequency parameters (a Japanese sentence of A01 in the ATR-503 data set).

4. Speech Synthesis Experiment and Evaluation

4.1. Evaluation criterion

To evaluate the spectral distortion caused by the training process, we measured the spectral distortion between the synthesized speech and real speech.

Bark spectral distortion [11] is known to be an objective measure for incorporating psychoacoustic responses. We used Bark spectral distortion and log spectral distortion to measure the spectral divergence between the synthesized and real speech. Using this criterion, we compared the present method with HTS-2.1 [10], a conventional HMM speech synthesis system with mel-cepstrum features and the global variance model [**?**].

The time alignment between the two spectral sequences was computed by using the dynamic time warping (DTW) algorithm. The average distortions on 53 synthesized speech sentences were compared and a statistical test was conducted.

4.2. Experimental Conditions

All the phoneme boundaries of the training data were given by the phoneme labels of HTS [10]. This means that the state sequences of the HMM are assumed to be known in the training stage. The number of the Gaussian functions in the CWM was set at 24. The initial CWM parameters were determined by performing the method reported by [9] to estimate the CWM parameters from the entire spectral sequence corresponding to each phoneme. We used an acoustic model of 1 state and a left-to-right HMM with no skip, as with HTS-2.1 [10]. We used "STRAIGHT" [12] to compute the spectral envelopes of the training data, with an analysis interval of 5 ms. We used 450 uttered sentences for the training and 53 uttered sentences for the synthesis and evaluation, respectively. All the speech samples were uttered by a Japanese male speaker and recorded with a 16kHz sampling rate and a 16 bit quantization, which were taken from the demo site of HTS [13].

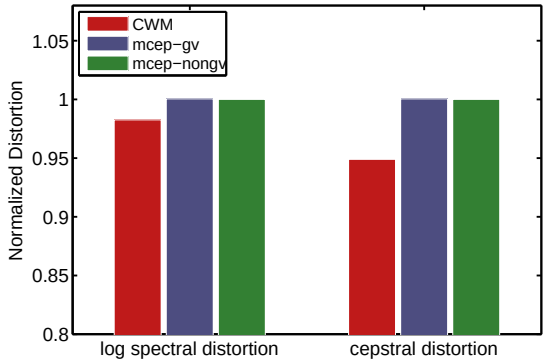


Figure 6: Normalized squared distances between the log spectra and cepstra of real and synthesized speech by the proposed method (red) , the ordinary mel-cepstral method with gv (blue) and without gv (green).

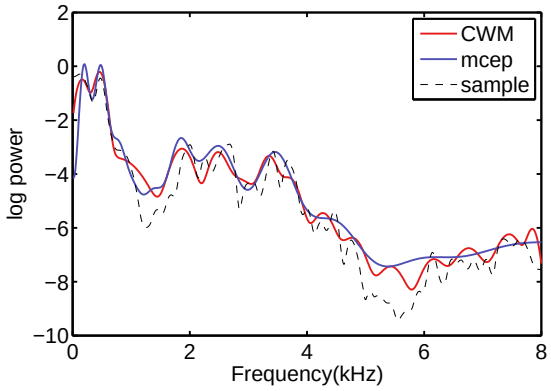


Figure 7: An example of the synthesized spectral envelope corresponding to the phoneme '/e/'

4.3. Experimental Results

The measured spectral distortions are shown in Fig. 6. The distances were normalized according to the values obtained with the proposed and conventional methods. We confirmed through a statistical test that the Bark spectral distortion obtained with the proposed method was significantly smaller than that obtained with the conventional method, while the two methods were comparable in terms of the log spectrum distortion. This result can be interpreted as showing that the proposed model was able to synthesize spectra properly especially in a low frequency region. Since the changes in the peak frequencies of spectra are mainly seen in the low frequency region, this result indicates that the proposed model is superior to the conventional model as regards modeling fluctuations in the frequency direction. Fig. 7 shows an example of the synthesized spectral envelope along with the spectral envelope of real speech corresponding to the phoneme '/e/'. In addition to the superiority of the proposed model in terms of the Bark spectral distortion, the proposed method was able to restore spectral peaks and dips more clearly than the conventional model, especially in the 1.5 to 6 kHz frequency range. In future we plan to conduct a subjective evaluation test to confirm whether the perceptual quality of the synthesized speech has actually been improved.

5. Conclusion

This paper proposed a text-to-speech synthesis (TTS) system based on a combined model consisting of the Composite Wavelet Model (CWM) and the Hidden Markov Model (HMM). To avoid the over-smoothing of spectra generated by the conventional HMM-based TTS systems using cepstral features, we considered it important to focus on a different representation of the generative process of speech spectra. In particular, we chose to characterize the speech spectra with the CWM, whose parameters correspond to the frequency, gain and peakiness of each underlying formant. This idea was motivated by our expectation that averaging these parameters would not directly cause the over-smoothing of the spectra, unlike the cepstral representations. To describe the entire generative process of a sequence of speech spectra, we combined the generative process of a formant trajectory using an HMM and the generative process of a speech spectrum using the CWM. We developed a parameter learning algorithm for this combined model based on an auxiliary function approach. We confirmed through experiments that our speech synthesis system was able to generate speech spectra with clear peaks and dips, which resulted in natural-sounding synthetic speech.

6. References

[1] T. Yoshimura, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, "Simultaneous modeling of spectrum, pitch and duration in HMM-based speech synthesis," in *Proc. of Eurospeech 1999*, 1999, pp. 2347–2350.

[2] H. Zen, K. Tokuda, and T. Kitamura, "An introduction of trajectory model into HMM-based speech synthesis," in *Proc. of 5th ISCA Speech Synthesis Workshop*, 2004, pp. 191–196.

[3] Y. Qian, Z.-J. Yan, Y.-J. Wu, F. K. Soong, G. Zhang, and L. Wang, "An HMM trajectory tiling (HTT) approach to high quality TTS," in *Proc. of Interspeech 2010*, 2010, pp. 422–425.

[4] F. Itakura, "Line spectrum representation of linear predictor coefficients of speech signals," *J. Acoust. Soc. Am.*, vol. 57, no. 1, p. S35, 1975.

[5] T. Toda and T. Tokuda, "A speech parameter generation algorithm considering global variance for HMM-based speech synthesis," *IEICE Trans. Inf. & Syst.*, vol. E90-D, no. 5, pp. 816–824, 2007.

[6] T. Saikachi, K. Matsumoto, S. Sako, and S. Sagayama, "Discussion about speech analysis and synthesis by composite wavelet model," in *Proc. ASJ Spring Meeting*, vol. 2, 2006, pp. 89–94, in Japanese.

[7] P. Zolfaghari and T. Robinson, "Formant analysis using mixtures of Gaussians," in *Proc. of ICSLP '96*, vol. 2, 1996, pp. 1229–1232.

[8] H. Kameoka, N. Ono, and S. Sagayama, "Speech spectrum modeling for joint estimation of spectral envelope and fundamental frequency," *IEEE Trans. Audio, Speech & Language Process.*, vol. 18, no. 6, pp. 1507–1516, 2010.

[9] N. Hojo, K. Minami, D. Saito, H. Kameoka, and S. Sagayama, "HMM speech synthesis using speech analysis based on composite wavelet model." in *Proc. ASJ Autumn Meeting*, no. 2-7-7, 2013, in Japanese.

[10] H. Zen, T. Nose, J. Yamagishi, S. Sako, T. Masuko, A. W. Black, and K. Tokuda, "The HMM-based speech synthesis system version 2.0," in *Proc. Proc. of 6th ISCA Speech Synthesis Workshop*, vol. 6, 2007, pp. 294–299.

[11] S. Wan, A. Sekey, and A. Gersho, "An objective measure for predicting subjective quality of speech coders," *IEEE Journal on Selected Areas in Communications*, vol. 10, no. 5, pp. 819–829, 1992.

[12] H. Kawahara, I. Masuda-Katsuse, and A. Cheveigne, "Restructuring speech representations using a pitch-adaptive time-freqency smoothing and an instantaneous-frequency-based f0 extraction," *Speech Communication*, vol. 27, no. 3–4, pp. 187–207, 1999.

[13] http://hts.sp.nitech.ac.jp/.

An experimental comparison of multiple vocoder types

Qiong Hu[1], *Korin Richmond*[1], *Junichi Yamagishi*[1,3], *Javier Latorre*[2]

[1]The Centre for Speech Technology Research,University of Edinburgh,U.K.
[2]Toshiba Research Europe Ltd, Cambridge, U.K.
[3]National Institute of Informatics, Tokyo, Japan

Qiong.Hu@ed.ac.uk, {korin,jyamagis}@inf.ed.ac.uk, javier.latorre@crl.toshiba.co.uk

Abstract

This paper presents an experimental comparison of a broad range of the leading vocoder types which have been previously described. We use a reference implementation of each of these to create stimuli for a listening test using copy synthesis. The listening test is performed using both Lombard and normal read speech stimuli, and with two types of question for comparison. Multi-dimensional Scaling (MDS) is conducted on the listener responses to analyse similarities in terms of quality between the vocoders. Our MDS and clustering results show that the vocoders which use a sinusoidal synthesis approach are perceptually distinguishable from the source-filter vocoders. To help further interpret the axes of the resulting MDS space, we test for correlations with standard acoustic quality metrics and find one axis is strongly correlated with PESQ scores. We also find both speech style and the format of the listening test question may influence test results. Finally, we also present preference test results which compare each vocoder with the natural speech.
Index Terms: Speech Synthesis, Vocoder, Similarity, Quality

1. Introduction

The prominence of the hidden Markov model (HMM) based approach to speech synthesis has grown rapidly in recent years, driven by its recognised advantages of convenient statistical modelling and flexibility. However, more than just convenient and adaptable speech synthesis alone, we desire the speech produced to be as close to natural speech as possible. For this, the characteristics of the speech vocoder used to generate the speech waveform from the vocoder parameters provided by the HMM are of paramount importance.

Various types of source-filter vocoder have typically been used for HMM-based speech synthesis Toolkit (HTS) [1] so far, where the excitation source is modelled by a mixture of pulse train and white Gaussian noise. Although the simplest pulse/noise model is straightforward, it does not provide an adequate model for the natural source and produces a characteristic "buzzy" sound due to strong harmonics at higher frequencies. Many more sophisticated source-filter vocoders have been proposed to address this problem. STRAIGHT [3] uses aperiodicity to weight the harmonic and noise components of the excitation. Substituting the pulse train with a residual signal is another way to retain a more detailed excitation signal, for example the Deterministic plus Stochastic Model [4]. Similarly, Glottal Inverse Filtering has been combined with HTS to model glottal pulses [5]. Meanwhile, multiple sinusoidal vocoders, have also been proposed. These depart from the strict source-filter approach to speech production, and generally differ in how they model the noise component. The Quasi-Harmonic Model [6] is an example of this sinusoidal class of vocoder.

Although a large number of good quality vocoders have been proposed, the optimal choice of vocoder to use in an HMM-based TTS system has not yet been clearly established. There are two main reasons for this. First, studies introducing a new vocoder are often limited to using a single baseline in the experimental validation they present. Second, when introducing a new vocoder, attention is not always given to evaluating the suitability of the vocoder for TTS. To address this open question, we attempt here a systematic comparison of a variety of vocoder types, and consider their suitability for HMM-based synthesis systems. We can find some previous work with a similar aim, for example different types of vocoder are introduced in some detail in [7], but generally there has not been a great deal of work in this direction. The aim of this paper, then, is to evaluate different vocoders in a reasonably large-scale listening test, using the same speech data and under consistent controller experimental conditions. We then apply multi-dimensional scaling and K-means clustering to analyse and visualise the responses and explore the relationship between the different vocoders.

When interpreting the results of this comparison, it is necessary to bear in mind certain caveats. First, the performance of waveform vocoders (harmonic, quasi-harmonic, etc.) are not distinguished from other vocoders, which are more suited to TTS system modelling as they have a fixed low dimension parameters for each frame. Moreover, this experiment is just based on one single speaker and limited set of samples. Thus, every vocoder may not be equally stretched in all ways possible, and so a truly even comparison may not be achieved. Another difficulty arises in differences in the parameters used by each vocoder. As explained further in Section 2, rather than implement every vocoder, this study uses the authors' own implementation for some vocoders (specifically, those proposed by Degottex, Drugman, Erro and Raitio). This means some parameter settings (e.g. F0 tracking) may vary between systems, which will affect the results. Nevertheless, the results of this study may still offer useful insights in terms of: i) similarities and differences between vocoder types; ii) whether any parameters greatly affect speech quality; iii) which vocoders are most natural and which are most amenable to statistical modelling.

This paper is organised as follows. The vocoders selected for comparison are briefly summarised in detail in Section 2. A series of comparisons are analysed based on both subjective and objective experiments in Section 3. Some discussion and conclusions are listed in Section 4.

2. Vocoder systems

The vocoders included in the listening test are summarised in Table 1, where each vocoder's name, suitability and parameters for HTS modelling also shown. In terms of sinusoidal vocoders, Harmonic plus noise model(HNM) vocoder based on

Table 1: *Summary of selected vocoders (k: number of sinusoids per frame, HTS: the suitability for HTS modelling).*

Name	Vocoder	HTS	Parameters per frame
MGC	Mel - generalised cepstral vocoder	Yes	MGC: 24 + F0: 1 , Pulse plus noise excitation
SF	STRAIGHT with full band mixed excitation	No	Aperiodicity:1024, spectrum: 1024+ F0:1, Multi- band mixed excitation
SC	STRAIGHT-MGC with critical band mixed excitation	Yes	Band aperiodicity: 25 + MGC :39 + F0: 1, Multi-band mixed excitation
Glot	Glottal vocoder	Yes	F0:1, Energy:1, HNR: 5, Source LSF: 10, Vocal tract LSF: 30, natural pulse
DSMR	MGC vocoder with DSM-based residual	Yes	MGC: 30 + F0:1, DSM for residual excitation
HM	Harmonic model	No	2*k harmonics + F0:1, Harmonic excitation
HMF	Harmonic with fixed dimension	No	2*k harmonics + F0: 1, Harmonic excitation
HNM	HNM-MGC vocoder	Yes	MGC:40 + F0:1, Multi- band excitation, Maximum voiced frequency
aHM	Adaptive harmonic model	No	2*k + F0:1, Harmonic excitation
OS	Original speech		

mel-cepstral coefficients and F0 [9], adaptive harmonic vocoder [6], harmonic vocoder [8], harmonic vocoder with fixed parameters were selected. For the source-filter vocoders,the deterministic plus stochastic model for residual(DSMR) vocoder [4], the mel-generalized cepstral vocoder, glottal vocoder [5], and STRAIGHT [3] with both full-band and critical-band based mixed excitation [10] were chosen for comparison.

2.1. Mel-generalized cepstral vocoder (MGC)

Here, a simple pulse/noise excitation is used for the MGC vocoder. Although straightforward, this excitation model cannot fully represent natural excitation signals and often generates "buzzy" speech. Different types of coefficients may be used to represent the spectrum. Mel-cepstra are often used, providing a good approximation to the human auditory scale of speech. Here, we use the Mel-Generalised Log Spectral Approximation (MGLSA) digital filter to filter the excitation signal to synthesise speech. We use the same parameter value in [4] α=0.42 and γ=-1/3 for MGC extraction.

2.2. STRAIGHT with full-band mixed excitation (SF)

STRAIGHT (Speech Transformation and Representation using Adaptive Interpolation of Weight Spectrum) [3] was developed to better remove the periodicity effects of F0 on extracting the vocal tract spectral shape. For spectral envelope extraction, both F0 adaptive spectral smoothing and compensatory time windows are used to transfer the time frequency-smoothing problem to frequency domain. Aperiodicity of the signal is computed as the difference between the upper and lower envelope of the spectrum. For voiced frame, noise is calculated by modulating the randomness of the phase component according to aperiodicity. Finally, all parameters are sent to a minimum-phase filter with group delay phase manipulation to synthesise speech.

2.3. STRAIGHT mel-generalised cepstal vocoder with critical band mixed excitation (SC)

Although STRAIGHT uses both aperiodicity and F0 adaptive spectral smoothing to solve the "buzzy" problem, the number of parameters for both the spectrum and aperiodicity components is the same size as the FFT length used, which is not suitable for statistical modelling. [10] proposed to use other lower dimensional parameters, such as Mel-generalized Cepstral Coefficients or Line Spectral Pairs to represent the spectrum instead. Here, in order to compare with other vocoders with similar spectral parameters, the Mel-generalised cepstral is chosen as the intermediate spectral parameterization. Aperiodicity parameters are also compressed by averaging the whole points to 25 sub-bands. The same type of filter is chosen as used in the STRAIGHT vocoder above.

2.4. Glottal vocoder (Glot)

[5] proposed a method to represent the glottal pulse signals instead of using a pulse-train excitation to represent the voiced excitation. For voiced speech frames, Interactive Adaptive Inverse Filtering (IAIF) is used to separate the glottal source from the vocal tract so that both the vocal tract and source signal may be accurately estimated. For unvoiced frames, conventional inverse filtering is applied. Other parameters, such as energy and harmonic-to-noise ratio (HNR), are calculated so as to weight the noise component of the source. During synthesis, a prestored library pulse is selected and interpolated to match the target F0. The glottal spectrum, HNR and energy also have to be set to match the target. Finally, a vocal tract filter as derived from analysis part is applied to the excitation to generate the speech signal.

2.5. MGC vocoder with DSM-based residual (DSMR)

In [4], a MGC vocoder with Deterministic plus Stochastic Model for residual signal is proposed. The residual signal is first obtained by applying inverse filtering using mel-generalised cepstrum filters. Then, a Blackman window, centred on glottal closure instants and of length equalling two F0 periods, is applied to obtain pitch-synchronous residual frames. In order to model these, they are first length normalised, then the deterministic component at the lower frequencies is decomposed using Principal Component Analysis (PCA) to obtain the first eigen residual. The energy envelope and an autoregressive model are used for the stochastic component. During synthesis, both these parts are resampled to match the target pitch to produce the new residual signal, which is used to drive a MGLSA filter to generate speech, so it is not strictly a sinusoidal vocoder.

2.6. Harmonic vocoder (HM)

Although real amplitude for the sinusoids were used for calculating parameters in [11], complex amplitudes proposed by [8], estimated by an algorithm operating in the time domain, are used in our experiment here, as it is easier to deal with the phase information (e.g. we can avoid problems such as phase unwrapping). For voiced frames, we calculate the complex amplitude by minimising the error between the original and estimated speech signals. The number of harmonics k in each frame is dictated by Fs/F0 (Fs: sampling frequency, F0: fundamental frequency). For unvoiced parts, Karhunen-Loeve expansion [12] shows we can use the same analysis as for voiced frames. We suppose that the frequency are close enough and set the F0 as 100Hz under the window length of 20ms to make the power spectrum change more slowly. From the complex amplitudes for a sequence of frames, we use the standard overlap and

add technique to re-synthesis speech.

2.7. Harmonic vocoder with fixed dimension(HMF)

From the description of the Harmonic Vocoder in the previous section, note the number of complex amplitude values in each frame varies depending on F0. This varying number of parameters is not suitable to combine with HTS. So, we also include a variant of the previous Harmonic vocoder in our experimental comparison that uses a fixed number of parameters per frame, which is labelled the "HMF" vocoder. To fix the number of harmonics, one option is to use those harmonics in at lower frequencies and add noise at higher frequencies. However, dividing the spectrum into two in this way would be rather arbitrary. For unvoiced speech in the "HM" vocoder, the number of harmonics in each frame is fixed, even though there may be no harmonics in fact. Similarly, here we suppose that the number of harmonics is the same as used for unvoiced parts irrespective of whether there are harmonics at higher frequencies or not.

2.8. HNM-MGC Vocoder (HNM)

A harmonic/stochastic waveform generator is presented by [9]. This method is based on the decomposition of the speech frames into a harmonic part and stochastic part and uses MGC, F0 and maximum voiced frequency (MVF) as an intermediate parameterization. This vocoder is thus suitable for statistical modelling with a fixed frame size. For voiced frames,the entire spectral envelope may be obtained by interpolating the amplitudes at harmonic points. Cepstral coefficients are obtained from the log spectrum and then they are reduced in number [2] and warped to the mel scale. Unvoiced part is just analysed through a fast Fourier transformation (FFT) and no stochastic part is assumed during analysis. MVF is caculated based on sinusoidal likeness measure. During synthesis, the cepstral envelope is resampled according to the harmonic points. Noise component is obtained by sampling the cepstral envelope at frequency above MVF. Minimum phase is using here.

2.9. aHM-AIR vocoder (aHM)

For the "HMF" and "HM" vocoders, we represent the whole band with harmonics alone. In principle, though, small errors in F0 value could cause large mismatch error in the higher frequencies. In order to solve this problem, [6] proposes a full-band adaptive harmonic vocoder without using any shaped noise. For analysis, it uses an Adaptive Iterative Refinement (AIR) method and an adaptive Quasi-Harmonic vocoder (aQHM) as an intermediate model to iteratively minimise the mismatch of harmonic frequencies while increasing the number of harmonics. Then, instantaneous amplitude and phase values may be obtained by interpolation. During synthesis, the aHM-AIR vocoder could represent the same structure by using only F0 rather than a frequency value at each analysis instant.

3. Experiment

3.1. Subjective analysis

Our approach to comparing and analysing the vocoders summarised in Section 2 relies upon multi-dimensional scaling (MDS)[14]. This technique aims to map points within a high dimensional space to a lower dimensional space while preserving the relative distances between the points. We can exploit this to visualise relative distances between the vocoders which indicate similarity in terms of perceptual quality. Listeners are asked to judge whether a given pair of stimuli are the same in terms of quality or different. Comparing a number of stimuli synthesised by all vocoders in this way, we obtain a matrix of inter-vocoder distance scores. This high-dimensional similarity matrix can be reduced to a 2- or 3- dimensional space to visualise vocoder similarities in terms of listener perception. The "Classical MDS" variant is used here, as we are comparing the Euclidean distance between each vocoder. Note we have found the natural speech is perceived as quite different from the vocoded speech, so including natural stimuli can heavily distort the relative distances between each vocoder if included. Therefore, we have omitted it from our MDS analysis. Instead, preference tests are subsequently used in order to compare the quality of each vocoder against the original speech.

Table 2: *Parameters for each section*

Section	Speaking style	Questions	ratio
1	Normal	Similarity	0.7943
2	Lombard	Similarity	0.7760
3	Normal	Preference	0.7500
4	Lombard	Preference	0.7451

In the test, every vocoder is compared pairwise with all others, giving a 9*9 similarity matrix. Phonetically balanced speech data from a UK male speaker is used for copy synthesis with each vocoder. The sampling rate is 16kHz. A total of 32 normal speaking style sentences and another 32 different sentences with Lombard speaking style are used. Several samples are available on the webpage (http://homepages.inf.ed.ac.uk/s1164800/vocoder_com.html). For each comparison unit and each listener, sentences are randomly selected for the matrix. So, all possible sentences could be heard for each comparison to mitigate sentence-dependent effects. Forty one native English speakers participated in the listening test, conducted in perceptual sound booth with headphones. Moreover, we suspect that the questions used for the listening test (same/different or better/worse/same) and the type of sentences (Lombard or Normal) could affect the MDS result as well. So, four sections are designed to test for this effect. A summary of the speaking styles, questions for comparing sentences and the eigenvalues ("ratio") for the first two dimensions found by MDS analysis are listed in Table 2.

The two-dimensional MDS spaces for the four test sections are shown in Figure 3. At first sight, it seems the locations of the vocoders differ in each section. However, by comparing the four MDS figures, we can see that although the absolute x- and y-coordinates for each point may vary, the relative positions of each vocoder are similar. The approximate consistency between the 4 different test sections indicates the relative layout of the vocoders observed is to some extent general, and that sufficient and adequate test stimuli have been selected, for example.

Next, we aim to analyse and interpret the relative layout of the vocoder points in the MDS space. Different speaking and question styles are used in each test section, and so we use Analysis of Variance (ANOVA) to ascertain whether these factors explain the variations observed. The results of both one-way and two-way ANOVAs are shown in Table 3. For the one-way method, the F-values for both speaking and question style for MDS are high. Meanwhile, both significances are less than 5 percent, which means these two factors greatly affect listener judgement. The two-way ANOVA indicates there is no significant interaction between the effects of speaking style and question type on listener judgement. We conclude therefore that speaking style and question format to some extent explain why each section map differs. Furthermore, in Table 2, note the ratio for the "same/different" question type is higher than that ob-

Table 3: *ANOVA for speaking style and question type*

Type	Anova	F value	Significance
One-way	Data~Style	6.7775	0.00993
One-way	Data~ Question	18.659	2.471e-05
Two-way	Data~Style*Question		
	Style	7.3651	0.007243
	Question	19.1647	1.949e-05
	Style:Question	0.0006	0.980126

tained used the 3-way "better/worse/same" question type. We believe therefore the first question type may yield more dependable results. So, for objective analysis, only section 1 and 2 are used for Normal speech and Lombard speech separately.

Although proximity in the MDS map can be interpreted as similarity, the relationship between the vocoders is not yet necessarily clear, so it would be more obvious to merge similar vocoders together. Thus, based on the 9*9 matrix of Euclidean distance between each vocoders, we use K-means clustering to identify emergent groupings. The "Silhouette" value [13] for varying numbers of clusters is computed, and the highest value is taken to indicate the optimum cluster number. The result for each test section is shown in Figure 4. The MDS results show that the SC, SF, MGC and Glot vocoders are very close to each other, indicating listeners find they sound similar to one another. A similar situation is observed for the DSMR and HNM vocoders, and for the aHM and HM vocoders. The clustering result in Figure 4 is consistent with this. In test section 1, except DSMR which uses DSM for residual siganl but is still based on source-filter model, vocoders in cluster two (in red) all use harmonics to describe speech. It is interesting that they all cluster separately from cluster one (in blue), where the vocoders belong to the traditional source-filter paradigm. More specifically, SC is merely a reduced dimension version of SF. Meanwhile, the intermediate parameters transferred from spectrum is the Mel Generalized Cepstrum, so it is also reasonable for MGC vocoder to be close to SF and SC. For other test sections, the situation is similar except for the relative change of the HM and HMF vocoders. Thus, we conclude that in terms of quality, the sinusoidal vocoders in this experiment sounds quite different from source filter vocoders, and there may be other reasons for DSMR clustering together with sinusoidal vocoders.

Having established similarities between vocoders, we also assess their relative quality compared to natural speech. A preference test is conducted for this purpose. Thirty two normal sentences and another 32 Lombard speech are surveyed separately. The same 41 native listeners participated in this test to give their preference in term of quality. The results given in Figure 1 show that the sinusoidal vocoders give relatively good quality. To further analyse the robustness of each vocoder for modelling both Normal and Lombard speech, the difference in preference scores between these 2 speech styles is presented in table 4. As we can see, in general, sinusoidal vocoders like HMF, HM and aHM give much less variable performance than the source/filter vocoder type. Interestingly, the SF vocoder gives stronger performance in terms of listener preference for Lombard speech than it does for normal speech in Figure 1. The reason for this is the subject of ongoing research.

3.2. Objective analysis

In this section, we explore why the vocoders cluster together as observed and what potential factors underpin listener judgements. A range of standard acoustic objective measures are calculated:

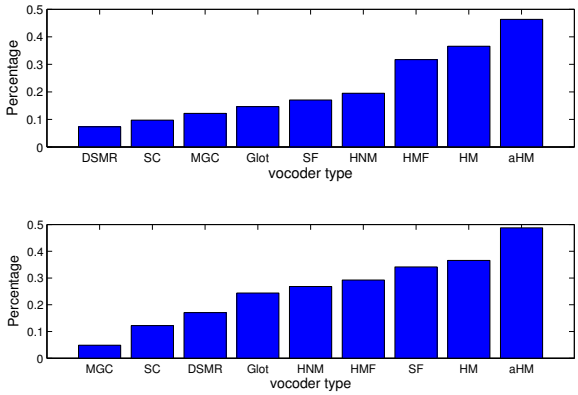


Figure 1: *Preference Test Result (up: Normal, down: Lombard)*

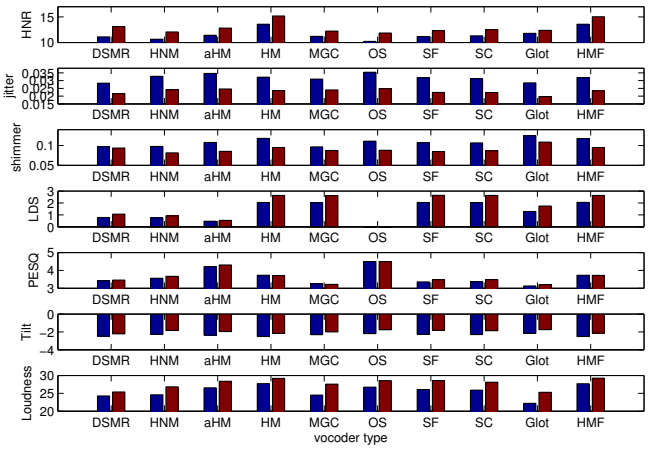


Figure 2: *Objective value result (blue: Normal , red: Lombard)*

- HNR (Noise Harmonic Ratio)
- Jitter
- Shimmer
- LDS (Log distance of spectra using FFT)
- PESQ (Perceptual Evaluation of Speech Quality)
- Spectral Tilt
- Loudness (Based on Model of ISO 532B)

The mean values for these acoustic measures are shown in Figure 2. Unfortunately, we can find no obvious relationship between these measures and the distances between the different vocoders. We attempt to interpret the significance of the MDS map axes by using linear regression and stepwise regression between the two axes and the given acoustic measures. As space is limited here, only the measure most highly correlated with the axes is listed in Tables 5.

As Table 5 shows, the significance of the correlation between PESQ scores with one axis of the MDS map is strong. In fact, combined with Figure 2, we can track vocoder quality through the axis value in MDS to a certain degree. For example, in test section 1 for normal speech, lower x-coordinates indicate higher quality in the vocoder. A similar situation applies

Table 4: *Vocoder preference stability result (Lombard preference value minus that for normal speech)*

vocoder type	DSMR	HNM	aHM	HM	MGC	SF	SC	Glot	HMF
preference vaule (Lombard - Normal)	0.0976	0.0732	0.0244	0	-0.0732	0.1707	0.0244	0.0976	-0.0244

Table 5: *linear regression result.*

linear regression	Significance	R squared
Section1_x~PESQ	0.00174	0.7746
Section2_x~PESQ	0.00991	0.6372

to Lombard speech in test section 2. The aHM vocoder has the best quality, followed by the HM vocoder. Note, though, that neither of these are currently suitable for statistical modelling. For the source-filter vocoders, the Glot, SF and SC ones all sound much better than MGC, and they are suited to modelling as well. Of the sinusoidal vocoders, not only are the HNM and DSMR vocoders suitable for modelling, but also appear to give good vocoded speech quality. The HMF vocoder also appears effective for producing speech with a fixed number of parameters. Finally, we consider which acoustic feature may be most related with other MDS axis. Unfortunately, there is no apparent pattern between any acoustic measure and the axis in the stepwise multi-linear regression. Therefore, we conclude that the listener perception judgements may be a more complex combination of multiple potential features.

4. Discussion and conclusion

This paper examines a broad range of vocoders and presents an experimental comparison to evaluate their relationship and potential factors that correlate with perceived vocoder quality. Both Lombard and normal read speech are used as stimuli produced by copy synthesis with each vocoder. MDS is conducted on the listener responses to analyse similarities in terms of quality between the vocoders. Four combinations of speaking style and listening test question format are tested. ANOVA results shows both speaking style and question format greatly affect listener judgements. For the preference question type, the eigenvalues for the first two dimensions in MDS space are somewhat reduced. Thus, we deem the similarity question type is more suitable for MDS analysis, and Lombard and Normal speech are surveyed separately in the subsequent analysis. Comparing preference test results for Normal and Lombard speech, we also find that sinusoidal vocoders give more consistent performance than source filter vocoders.

To analyse their potential relationship in more depth, K-means clustering is applied to the listener similarity judgment matrix and combined with the MDS results. We find in terms of quality, the sinusoidal vocoders cluster separately from the source filter vocoders. Thus, we conclude that sinusoidal vocoders are perceptually distinguishable from source filter ones. The preference test comparisons with the natural stimuli presented here indicate sinusoidal vocoders can give superior vocoded speech quality. In order to interpret the axes of the obtained MDS space, a several objective acoustic measures are tested for correlation with the MDS space axes. Linear regression result shows that one axis is related with quality. However, no obvious acoustic measure could be found to explain the other axis of the two dimensional MDS space, which we interpret as implying that human perception of vocoded speech quality may combine multiple factors.

5. Acknowledgements

This research is supported by Toshiba. The authors also greatly appreciate help from Gilles Degottex (University of Crete), Tuomo Raitio (Aalto University), Thomas Drugman (University of Mons) and Daniel Erro (University of the Basque Country) by generating samples from their vocoder implementations.

6. References

[1] H. Zen, T. Nose, J. Yamagishi, S. Sako, T. Masuko, A. Black, and K. Tokuda, "The HMM-based speech synthesis system (HTS) version 2.0." Proc. of Sixth ISCA Workshop on Speech Synthesis. 2007, pp. 294–229.

[2] K. Tokuda, T. Kobayashi, T. Masuko and S. Imai, "Melgeneralized cepstral analysis — A unified approach to speech spectral estimation," Proc. ICSLP'94, pp.1043– 1046, Yokohama, Japan, Sep. 1994

[3] H. Kawahara, I. Masuda-Katsuse, and A. de Cheveigne, "Restructuring speech representations using a pitch-adaptive time-frequency smoothing and an instantaneous-frequency-based F0 extraction: Possible role of a repetitive structure in sounds," Speech Communication, vol.27, pp.187-207 (1999).

[4] T. Drugman and T. Dutoit, "The deterministic plus stochastic model of the residual signal and its applications," IEEE Trans. Audio, Speech and Language Processing, vol. 20, no. 3, pp. 968–981, March 2012.

[5] T. Raitio, A. Suni, J. Yamagishi, H. Pulakka, J. Nurminen, M. Vainio, and P. Alku, "HMM-based speech synthesis utilizing glottal inverse filtering," IEEE Trans. on Audio, Speech, and Lang. Proc., vol. 19, no. 1, pp. 153–165, Jan. 2011.

[6] G. Degottex and Y. Stylianou. "A Full-Band Adaptive Harmonic Representation of Speech."In Proc. Interspeech, Portland, USA. ISCA, September 2012.

[7] M. Airaksinen. "Analysis/Synthesis Comparison of Vocoders Utilized in Statistical Parametric Speech Synthesis." Master thesis, Aalto University, November 2012

[8] Y. Stylianou, "Harmonic plus Noise Models for Speech, combined with Statistical Methods, for Speech and Speaker Modification", Ph.D. thesis, Ecole Nationale Superieure des Telecommunications, January 1996.

[9] D. Erro, I. Sainz, E. Navas, I. Hernaez, "Improved HNM-based Vocoder for Statistical Synthesizers", Proc. Interspeech, pp. 1809-1812, Florence, August 2011.

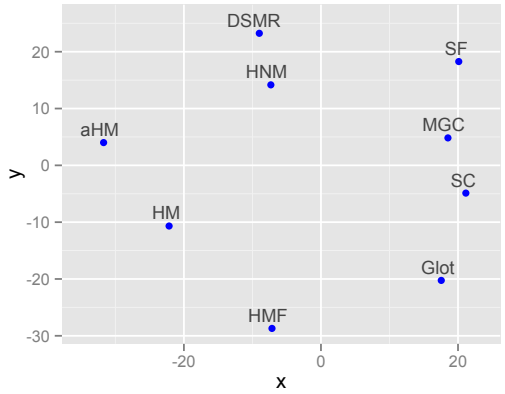
[10] H. Zen, T. Toda, M. Nakamura, and T. Tokuda, "Details of the Nitech HMM-based speech synthesis system for the Blizzard Challenge 2005," IEICE Trans. Inf. & Syst., vol. E90-D, no. 1, pp. 325–333, 2007.

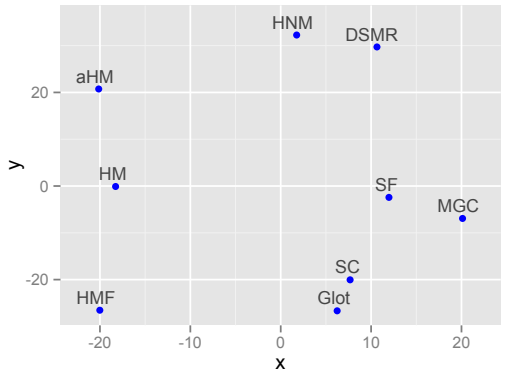
[11] S. Shechtman, and A. Sorin. "Sinusoidal model parameterization for HMM-based TTS system." In Proc. Interspeech, pp.805-808., Makuhari, Japan, September 2010.

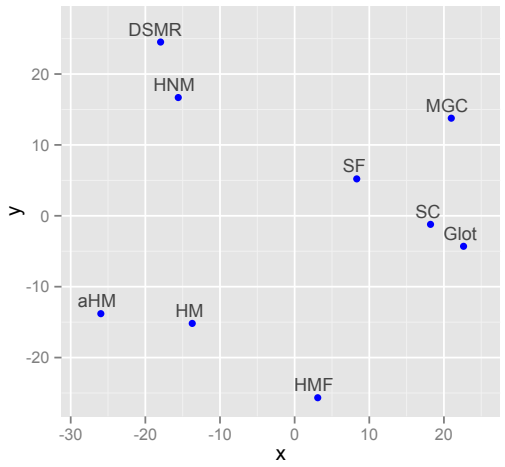
[12] Quatieri, F. T., "Discrete time speech signal processing", Pearson education, 427-439, 2004.

[13] P. Rousseeuw. "Silhouettes: a graphical aid to the interpretation and validation of cluster analysis." Journal of computational and applied mathematics 20 (1987): 53-65.

[14] C. Mayo, R. A. J. Clark, and S. King."Multidimensional scaling of listener responses to synthetic speech." In Proc. Interspeech 2005, Lisbon, Portugal, September 2005.







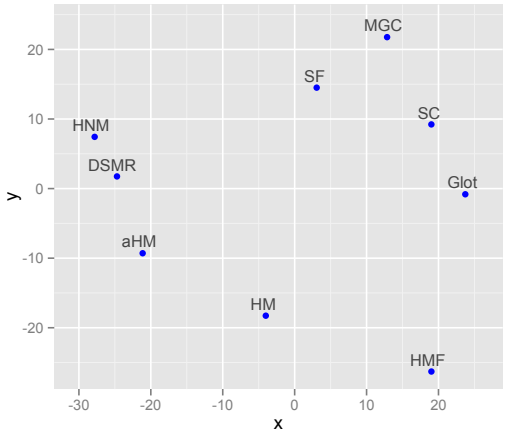
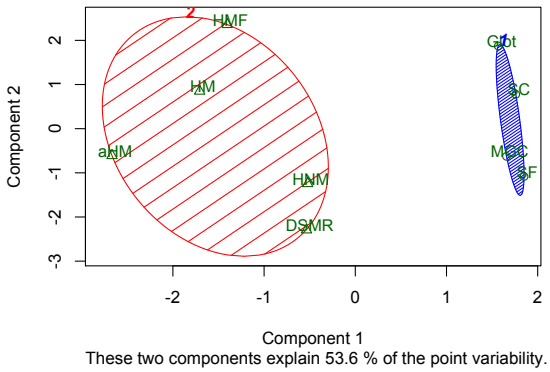
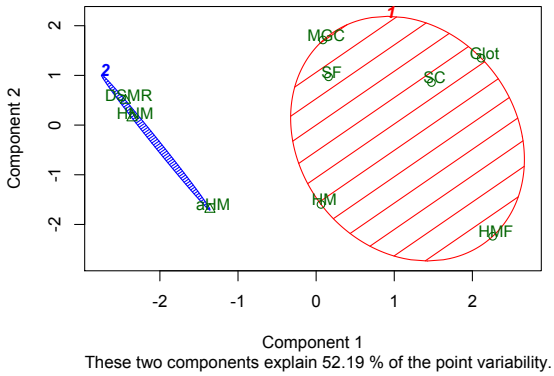
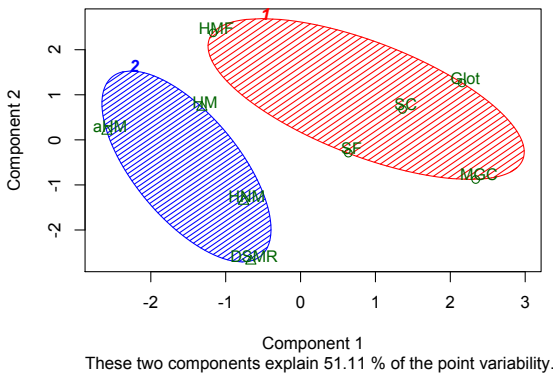


Figure 3: *MDS results for each section(up to down 1,2,3,4)*







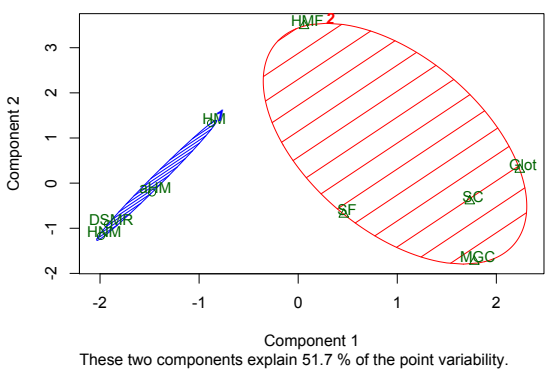


Figure 4: *K-means clustering results for each section (up to down 1,2,3,4)*

Statistical Model Training Technique for Speech Synthesis Based on Speaker Class

Yusuke Ijima, Noboru Miyazaki, Hideyuki Mizuno

NTT Media Intelligence Laboratories, NTT Corporation, Japan

Abstract

To allow the average-voice-based speech synthesis technique to generate synthetic speech that is more similar to that of the target speaker, we propose a model training technique that introduces the label of *speaker class*. Speaker class represents the voice characteristics of speakers. In the proposed technique, first, all training data are clustered to determine classes of speaker type. The average voice model is trained using the labels of conventional context and speaker class. In the speaker adaptation process, the target speaker's class is estimated and is used to transform the average voice model into the target speaker's model. As a result, the speech of the target speaker is synthesized from the target speaker's model and the estimated target speaker's speaker class. The results of an objective experiment show that the proposed technique significantly reduces the RMS errors of log F0. Moreover, the results of a subjective experiment indicate that the proposal yields synthesized speech with better similarity than the conventional method.

Index Terms: HMM-based speech synthesis, average voice model, speaker adaptation, speaker clustering

1. Introduction

Recent research on text-to-speech synthesis has focused on supporting arbitrary speakers given only a small amount of the target speaker's speech data. In HMM-based speech synthesis systems [1], the average-voice-based speech synthesis technique with model adaptation [2] has been proposed. Given only a few minutes of the target speaker's speech data, this technique can synthesize arbitrary texts by transforming the average voice model to the target speaker's model.However, it has been reported that the similarity of the synthesized speech to the target speaker's speech is degraded by model conversion if the acoustic features of the average voice model are distant from those of the target speaker [3]. One useful solution is creating an average voice model whose characteristics are close to those of the target speaker.

To realize this approach, a similar speaker selection based model training technique has been proposed [4]. In this technique, synthetic speech is made closer to that of the target speaker by training an average voice model from perceptually similar speakers (manually selected); note that speaker selection decreases the amount of training data. However, in the case of automatic similar speaker selection using acoustic features, it was reported that the similarity of synthesized speech is degraded due to this reduction. Although, these results indicate that the selection must identify perceptually similar speakers to improve the similarity of the synthesized speech, it is well known that this selection is very difficult. To avoid these problems, i.e., selecting perceptually-similar speakers and offsetting the decrease in amount of training data, it is desirable to create one average voice model that can take into account of multiple speaker characteristics with no decrease in the amount of training data.

So that model training can take into account of the various characteristics of the training data, some studies have proposed a model training technique that adds characteristics of training data to the usual context set of phonetic, prosodic, and linguistic features. [5] proposed a style-mixed modeling technique that utilizes speaking styles and emotional expressions as context. In addition, the gender-mixed modeling technique, which uses speaker gender as an additional context, was proposed to enhance average-voice-based speech synthesis, and its effectiveness was shown [6]. In this study, we propose to add *speaker class*, which better represents detailed speaker characteristics than gender, to the average-voice-based speech synthesis technique.

In the proposed technique, a speaker clustering technique is applied to the training data so as to group the acoustic features of all speakers used for average voice model training. The average voice model is trained using the label of speaker class. In the speaker adaptation process, the target speaker's speaker class is estimated, and the average voice model is transformed to the target speaker's model using the labels of conventional context and the estimated speaker class. The key to realizing our proposal is the robust estimation of the target speaker's class. If complex features that have high correlation with perceptual similarity are used for speaker clustering, we would face the same problem of perceptually-similar speaker selection as in [4]. To avoid this problem, we use very simple acoustic features of spectrum, F0, and phoneme duration for speaker clustering and speaker class estimation. Objective and subjective evaluations show the effectiveness of the proposed technique.

2. Speech synthesis system with speaker class label

2.1. Overview of proposed speech synthesis system

A block diagram of the proposed speech synthesis method is shown in Fig. 1. The proposed technique first trains an average voice model using training data labeled with speaker class and other conventional contexts. The speaker class of the target speaker is estimated from the speaker's training data and input to the average voice model to transform it to better suit the target speaker. Given the estimated target speaker's class and other conventional contexts, the target speaker's model synthesizes the target speaker's speech. The overall process of training, adaptation, and speech synthesis is summarized below.

Training part:

Step 1 Apply the speaker clustering technique to all training data and define a finite number of speaker classes.

Step 2 Train an average voice model using the training data

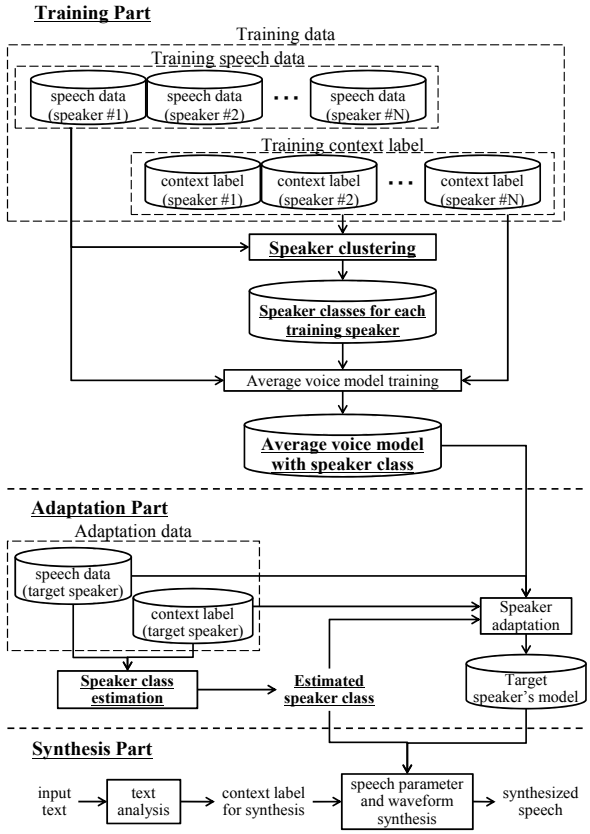


Figure 1: *Block diagram of the speech synthesis system.*

with labels of speaker class and conventional contexts.

Adaptation part:

Step 3 Estimate the speaker class of the target speaker using adaptation data.

Step 4 Transform the trained average voice model into the target speaker's model using the adaptation data and the estimated speaker class.

Synthesis part:

Step 5 Generate the context label from the result of text analysis.

Step 6 Generate the speech parameter sequence of the target speaker using the target speaker's model, the estimated speakers class and the generated context label.

Step 7 Synthesize the speech waveform of the target speaker.

In the proposed technique, if speaker class is estimated correctly, the leaf nodes that have similar speech characteristics to those of the target speaker are used for speaker adaptation and speech synthesis. Therefore, the output of the proposed technique is closer to the target speaker than is possible with the conventional technique. Details of this technique are described below.

2.2. Speaker class

We apply a speaker clustering technique to cluster the acoustic features of the speakers in the training data. For this, it might be thought necessary to use acoustic features that are highly correlated with perceptual similarity. Many previous studies showed that perceptual similarity is influenced by prosodic features, consisting of F0 and phoneme duration, and acoustic features, consisting of cepstral coefficients and the aperiodic component [4, 7–9]. However, since the key to realizing this approach is estimating the speaker class of the target speaker robustly, such complex features are not desirable.

In this study, in order to robustly estimate the speaker class of the target speaker, we utilize three simple features, average mel-cepstral coefficients, average logarithmic F0 (log F0), and speaking rate; they represent the features of spectrum, F0, and phoneme duration respectively. Speaker clustering, based on the k-means algorithm, is performed for each of the three features in isolation. The three acoustic features are described as follows.

2.2.1. Average mel-cepstral coefficients

Average mel-cepstral coefficients of all training data are used for spectrum-based speaker clustering. Because spectrum-oriented speaker characteristics are chiefly presented by voiced phonemes rather than unvoiced phonemes, the average mel-cepstral coefficients are obtained from only voiced frames as detected by TEMPO [10].

2.2.2. Average log F0

Average log F0 of all training data are used for F0-based speaker clustering. As per Sect. **2.2.1**, the average log F0 was obtained from only voiced frames as detected by TEMPO [10].

2.2.3. Speaking rate

Average speaking rates of all training data are used for phoneme-duration-based speaker clustering. The speaking rate is obtained from manually segmented phoneme boundaries of all training data. The speaking rate of speaker i (SR_i) is given by

$$SR_i = \frac{Mora_i}{UttLen_i} \quad (1)$$

where, $Mora_i$ and $UttLen_i$ represent, respectively, the number of mora of speaker i and the utterance length of speaker i.

2.3. Context clustering using speaker class label

Generally, in the average voice model training, decision tree-based context clustering for each model, i.e., mel-cepstrum, log F0, and phoneme duration, is performed using common questions. However, since our proposal adds speaker class to the other conventional contexts, using common questions may lead to negative effects on the tree structure. To avoid this problem, we also use model-specific questions. For instance, questions intended to identify the speaker class (speech rate) are used for context clustering for the phoneme duration model only. In this paper, the context clustering was performed before SAT.

2.4. Estimating speaker class of target speaker

To estimate the speaker class of target speaker, we use the very simple approach of Euclidean distance between the target speaker's features and the centroids of all clusters. Given the adaptation data of the target speaker, we first obtain the three features for the speaker, i.e., the average mel-cepstral coefficients, average log F0, and average speaking rate. Finally, the

Table 1: *The number of leaf nodes of decision trees for each feature.*

# of speaker class	model		
	mel-cepstrum	log F0	duration
1 (conventional)	4954	20941	2971
2	5260	29454	3054
4	5766	35120	2939
8	6607	37952	2952

Table 2: *Mel-cepstral distortion [dB] between original and synthetic speech for each target speaker (closed target speaker).*

# of speaker class	target speaker				
	#1	#2	#3	#4	#5
1	5.78	**5.90**	**5.85**	5.28	**5.83**
2	5.76	5.93	5.87	5.27	5.92
4	**5.73**	5.93	5.91	5.27	5.85
8	5.75	5.96	5.90	**5.25**	5.87

Table 3: *RMS errors of log F0 [cent] between original and synthetic speech for each target speaker (closed target speaker).*

# of speaker class	target speaker				
	#1	#2	#3	#4	#5
1	203.3	216.3	218.8	135.8	182.1
2	202.9	209.4	211.2	133.6	174.1
4	187.1	**199.5**	**204.3**	131.9	172.8
8	**183.1**	203.7	212.2	**127.2**	**169.1**

Table 4: *RMS errors of phoneme duration [ms] between original and synthetic speech for each target speaker (closed target speaker).*

# of speaker class	target speaker				
	#1	#2	#3	#4	#5
1	25.18	24.28	25.16	30.15	24.66
2	24.43	24.37	25.47	29.46	25.42
4	**23.55**	23.08	25.24	**28.70**	**24.19**
8	23.56	**22.73**	**24.66**	29.54	26.07

three subclasses (one per feature) of the target speaker are estimated to be those that have the smallest Euclidean distance between the input feature and cluster centroids.

3. Experiments

3.1. Experimental conditions

In the following experiments, we used the speech data gathered from 88 non-professional Japanese female speakers'. This database contains about 120 phonetically balanced sentences for each speaker. The speakers' ages ranged from 18 to 39.

The sampling frequency of the speech was 16 kHz and the quantization bit rate was 16 bits. We used STRAIGHT analysis [10] for speech feature extraction, and extracted spectral envelope, F0, and aperiodic components. The analysis frame shift was 5 ms. The spectral envelope was then converted to mel-cepstral coefficients using a recursion formula. The aperiodicity feature was also converted to average values for five frequency sub-bands, i.e., 0–1, 1–2, 2–4, 4–6, and 6–8 kHz. As a result, the feature vector consisted of 40 mel-cepstral coefficients including the zeroth coefficient, log F0, and five-band aperiodicity values with delta and delta-delta coefficients. The total dimensionality was 138. We used a five-state left-to-right hidden semi-Markov model with no skip topology. The output distribution in each state was modeled as a single Gaussian density function, and the covariance matrices were assumed to be diagonal.

For training the average voice model, one hundred sentences uttered by 85 of the 88 speakers were used. For its adaptation to the target speaker, twenty sentences uttered by the target speaker were used as adaptation data. We used the combined technique of CSMAPLR and MAP adaptation as the speaker adaptation algorithm [11].

In order to evaluate the effectiveness of the proposed speaker class approach, we created 4 trained average voice models with different numbers of speaker class, 1, 2, 4, and 8. The average voice model with 1 class represents the conventional average voice model.

3.2. The number of leaf nodes in the decision trees

In order to confirm the impact of the speaker class proposal has on the model structure, we investigated the number of leaf nodes in the decision trees for each of the four models. Table 1 lists the number of leaf nodes for each average voice model and each acoustic feature. The number of leaf nodes for the aperiodic feature is not shown because speaker class context determined from the aperiodic feature is not used in speaker clustering. We can see that the number of leaf nodes increases as the number of speaker class increases except for phoneme duration. This is because the amount of training data for phoneme duration is smaller than that for the other two features.

Furthermore, from the decision trees of each model, speaker classes associated with average log F0 tended to be split at the node close to the root node of the tree. On the other hand, speaker classes associated with the two other features tend to be split at nodes close to leaf nodes.

3.3. Objective evaluation

To objectively evaluate the proposed technique, we measured the mel-cepstral distortion, the RMS errors of log F0, and the RMS errors of phoneme duration between original and synthetic speech. To evaluate the influence of speaker class estimation, we used two types of target speakers (closed and open target speakers). The five closed target speakers were among those used for average voice model training, and their speaker classes for speaker adaptation were given correctly. The three open target speakers were not included in the average voice model training, and their speaker classes were estimated automatically. These eight speakers have different speaker classes about at least one feature. Twenty sentences uttered by each target speaker and used as the reference data were not included in the average voice model training data or speaker adaptation.

Table 2–4 show, respectively, the mel-cepstral distortion, the RMS errors of log F0, and the RMS errors of phoneme duration for each closed target speaker and each average voice model. From these results, we can see that the RMS errors of log F0 and phoneme duration are decreased by increasing

Table 5: *Mel-cepstral distortion [dB] between original and synthetic speech for each target speaker (open target speaker).*

# of speaker class	target speaker #1	#2	#3
1	**5.39**	5.87	6.03
2	5.41	**5.82**	6.02
4	5.44	5.86	**5.99**
8	5.45	5.87	6.03

Table 6: *RMS errors of log F0 [cent] between original and synthetic speech for each target speaker (open target speaker).*

# of speaker class	target speaker #1	#2	#3
1	184.9	217.3	205.6
2	174.9	211.4	**190.2**
4	**169.2**	**202.7**	193.3
8	173.8	208.9	193.2

the number of speaker classes. This indicates that the proposed technique enhances the effectiveness of the model's tree structure for log F0 and phoneme duration. On the other hand, the mel-cepstral distortions were not directly impacted by speaker class. This is because the speaker class yielded by average mel-cepstral coefficients does not adequately represent spectrum-oriented speaker characteristics. Therefore, to suppress mel-cepstral distortion, we have to determine which feature can best represent the spectrum-based characteristics of the speaker.

Table 5–7 also show, respectively, the mel-cepstral distortion, the RMS errors of log F0, and the RMS errors of phoneme duration for each open target speaker and each average voice model. These results demonstrate tendencies similar to those from the closed speakers test. However, when the number of speaker class is 8, RMS errors of most target speakers were higher to those with 4 classes. This is considered to be due to over-training. Therefore, we used the model with 4 classes in the following subjective experiment.

3.4. Subjective evaluation

We conducted a XAB test to evaluate voice characteristics and prosodic features of the synthesized speech using the model adapted from the conventional average voice model and the proposed average voice model. All permutations of synthetic sentence pairs matching each target speaker were created and presented in both orders (XAB and XBA), to eliminate bias in the order of stimuli. The subjects were ten persons, and each was presented synthesized speech samples and then asked which sample was similar to the reference speech. The reference speech was synthesized by a STRAIGHT vocoder. As in the objective evaluation of the open speakers, we used three open speakers as the target speaker, and twenty sentences as the evaluation sentences.

Figure 2 shows the preference scores for each target speaker. We can see that the proposed technique has better performance than the conventional technique. This indicates that the proposed technique based on speaker class can synthesize speech that is closer to the target speaker than the conventional alternative even though only three simple acoustic features are used for speaker clustering. However, since no performance comparison that changed the acoustic feature used for the speaker clustering was performed, it is necessary to evaluate the performance of speech synthesis by using other acoustic features that have high correlation with perceptual similarity as shown in [4].

Table 7: *RMS errors of phoneme duration [ms] between original and synthetic speech for each target speaker (open target speaker).*

# of speaker class	target speaker #1	#2	#3
1	22.11	19.78	21.60
2	23.78	19.71	22.95
4	**21.87**	**18.72**	**19.59**
8	21.92	19.06	20.14

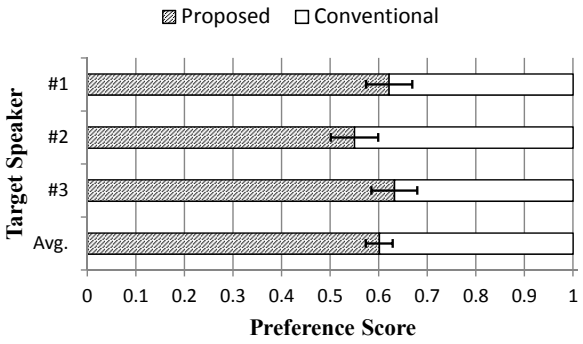


Figure 2: *Preference score from the subjective evaluation. (Error bars show the 95% confidence intervals.)*

4. Conclusion

In this paper, we proposed a model training technique that utilizes speaker class. This technique realizes robust speaker class estimation by using three simple features, the average mel-cepstral coefficients, average log F0, and speaking rate. Objective and subjective experiments showed that the proposed technique can synthesize speech that is closer to that of the target speaker than the conventional method. In particular, this technique can significantly reduce the RMS errors of log F0.

In future work, we will investigate other acoustic features and other speaker clustering techniques to improve the technique's speech synthesis performance. Applying the proposed technique to style adaptation [12] will also be investigated.

5. References

[1] H. Zen, K. Tokuda, T. Masuko, T. Kobayashi and T. Kitamura, "A Hidden semi-Markov model-based speech synthesis system," IEICE Trans. Inf. and Syst., vol.E90-D, no.5, pp.825–834, May. 2007.

[2] J. Yamagishi and T. Kobayashi, "Average-voice-based speech synthesis using HSMM-based speaker adaptation and adaptive training," IEICE Trans. Inf. and Syst. vol.E90-D, no.2, pp.533–543, Feb. 2007.

[3] J. Yamagishi, O. Watts, S. King and B. Usabaev, "Roles of the Average Voice in Speaker-adaptive HMM-based Speech Synthesis," in Proc. Interspeech 2010, pp.418–421, Sept. 2010.

[4] R. Dall, MC. Veaux, J. Yamagishi and S. King, "Analysis of speaker clustering strategies for HMM-based speech synthesis," in Proc. INTERSPEECH 2012, Sept. 2012.

[5] J. Yamagishi, K. Onishi, T. Masuko and T. Kobayashi, "Acoustic modeling of speaking styles and emotional expressions in HMM-based speech synthesis," IEICE Trans. Inf. & Syst., E88-D(3), pp.502–509, 2005.

[6] J. Yamagishi, T. Nose, H. Zen, Z. Ling, T. Toda, K. Tokuda, S. King, and S. Renals. "Robust speaker-adaptive HMM-based text-to-speech synthesis," IEEE Trans. Audio, Speech & Language Process., 17(6), pp.1208–1230, 2009.

[7] N. Higuchi and M. Hashimoto, "Analysis of acoustic features affecting speaker identification," in Proc. Eurospeech '95, pp.435–438, 1995.

[8] K. Amino, T. Sugawara and T. Arai, "Speaker Similarity in Human Perception and their Spectral Properties," in Proc. WESPAC IX, 2006.

[9] Y. Adachi, S. Kawamoto, S. Morishima and S. Nakamura, "Perceptual similarity measurement of speech by combination of acoustic features," in Proc. ICASSP 2008, pp.4861–4864, 2008.

[10] H. Kawahara, I. Masuda-Katsuse and A. Cheveigne, "Restructuring speech representations using a pitch-adaptive time-frequency smoothing and an instantaneous-frequency-based F0 extraction: Possible role of a repetitive structure in sounds," Speech Communication, 27, pp.187–207, 1999.

[11] J. Yamagishi, T. Kobayashi, Y. Nakano, K. Ogata and J. Isogai. "Analysis of Speaker Adaptation Algorithms for HMM-based Speech Synthesis and a Constrained SMAPLR Adaptation Algorithm," IEEE Trans. Audio, Speech & Language Process., 17(1), pp.66–83, 2009.

[12] M. Tachibana, J. Yamagishi, T. Masuko and T. Kobayashi, "A style adaptation technique for speech synthesis using HSMM and suprasegmental features," IEICE Trans. Inf. and Syst. vol. E89-D, no. 3, pp.1092–1099, Mar. 2006.

Is Intelligibility Still the Main Problem? A Review of Perceptual Quality Dimensions of Synthetic Speech

Florian Hinterleitner[1], *Christoph R. Norrenbrock*[2], *Sebastian Möller*[1]

[1]Quality and Usability Lab, TU Berlin, Germany
[2]Digital Signal Processing and System Theory, CAU Kiel, Germany
florian.hinterleitner@tu-berlin.de, cno@tf.uni-kiel.de,
sebastian.moeller@telekom.de

Abstract

In this paper, we present a comparative overview of 9 studies on perceptual quality dimensions of synthetic speech. Different subjective assessment techniques have been used to evaluate the text-to-speech (TTS) stimuli in each of these tests: in a semantic differential, the test participants rate every stimulus on a given set of rating scales, while in a paired comparison test, the subjects rate the similarity of pairs of stimuli. Perceptual quality dimensions can be derived from the results of both test methods, either by performing a factor analysis or via multidimensional scaling. We show that even though the 9 tests differ in terms of used synthesizer types, stimulus duration, language, and quality assessment methods, the resulting perceptual quality dimensions can be linked to 5 universal quality dimensions of synthetic speech: (i) *naturalness of voice*, (ii) *prosodic quality*, (iii) *fluency and intelligibility*, (iv) *disturbances*, and (v) *calmness*.

Index Terms: text-to-speech (TTS), perceptual quality dimensions, evaluation

1. Introduction

Even though the quality of modern TTS systems has reached a level of quality that no longer reminds listeners of robot-like voices but of real human speakers, different degradations still diminish the overall quality impression: most PSOLA-based diphone synthesizers lead to artificial voices due to frequent concatenations of speech units, HMM-synthesizers can generate natural-sounding but also "noisy" speech, and the quality of unit-selection systems mainly depends on the size of the used speech corpus, how well the units fit together and how well this corpus fits to the text that is to be synthesized. These impairments all sound differently, thus they degrade speech along different perceptual dimensions. Hence, the quality of synthetic speech is of multidimensional nature.

Several listening tests have been carried out over the past years in order to reveal the inherent perceptual quality dimensions of synthetic speech. As a result, a variety of different dimensions appear to exist. In one study [1] the dimensions were labeled (i) *prosody* and (ii) *segmental*, the next study [2] yielded the dimensions (i) *naturalness* and (ii) *intelligibility*, and another study [3] resulted in the dimensions (i) *naturalness of voice*, (ii) *temporal distortions*, and (iii) *calmness*. Given the different synthesizers that were used, the variations in stimulus duration, and the diverse assessment methodologies, the ambiguity is not surprising.

In this paper, we present a comparative overview of perceptual quality dimensions which resulted from 9 studies on TTS quality, and we will show that these dimensions can be attributed to a unifying set of dimensions. In Section 2, we introduce the two different approaches to multidimensional analysis for speech signals that were used in the 9 studies. Test details are given in Section 3. Section 4 highlights the similarities and differences between the studies. In Section 5, we compare the quality dimensions of all studies and introduce a set of 5 universal TTS-quality dimensions to which all other dimensions can be linked. Finally, in Section 6 we conclude the results and give a perspective to future work.

2. Multidimensional analysis

The two main approaches to analyzing perceptual quality dimensions with the help of human listeners are discussed in the following.

In a semantic differential (SD), pre-defined attribute scales are used to measure the auditory impression of the listeners. This guarantees a direct relationship between the used attribute scales and the derived quality dimensions. Therefore, the results are usually easy to interpret. On the downside, the ratings of the test participants are always limited to the set of presented scales. If a quality impression can not be expressed by any of the presented scales, this information will be lost. Thus, it is crucial to carefully choose a set of scales for the listening test. To reduce the influence of the test designers to a minimum, a suitable set of scales can be developed through several pretests, i.e., the goal of the first pretest is to collect attributes and corresponding attribute scales which describe the auditory impression of the listeners; in a second pretest, this set of attribute scales can be reduced to a final selection of scales.

In comparison, the multidimensional scaling (MDS) approach with paired comparison (PC) testing is solely based on the perceptual impression of the listener and not on any given rating scales. Participants are instructed to rate the similarity of one feature of pairs of speech signals, e.g., similarity in naturalness. Therefore, every stimulus in a set of n stimuli has to be compared to all remaining $n-1$ stimuli. The outcome is a matrix that represents the similarity between all stimuli [4]. Via an MDS algorithm, the dimensionality of this matrix can then be reduced until the solution is interpretable but still represents the observed stimulus space. However, since a complete comparison of all stimuli leads to $\frac{n(n-1)}{2}$ comparisons and a listening-test duration of several hours per subject, this approach is hardly deployable with larger sets of objects. For these cases, Tsogo [5] proposed a sorting task. Here, subjects are instructed to build groups of stimuli that are similar to each other while being different from the stimuli in other groups.

This yields one incidence matrix per subject from which a similarity matrix can be derived that can be further processed as described above. Even though the MDS approach has the advantage that the participants' ratings are not influenced by given rating scales, its major drawback is the interpretability of the resulting dimensions. MDS dimensions as such give no indication on their interpretation, thus, additional knowledge about the nature of the stimuli has to be obtained, e.g., via expert listening, rating scales or measures derived from the synthesis system.

3. Subjective TTS evaluations

This section gives an overview of the 9 different TTS databases as well as an interpretation of the resulting perceptual quality dimensions.

3.1. Test 1

In 1995, Kraft and Portele [1] evaluated five German-speaking TTS systems in an auditory listening test. The database consisted of stimuli produced by 2 formant synthesizers (male voices) and 3 diphone/demisyllable synthesizers (2 female voices, 1 male voice). The 44 subjects were instructed to rate the stimuli on 8 presented absolute category rating (ACR) scales with 5 to 6 categories. 6 familiar and unfamiliar passages were synthesized with a total duration of about 100 words. A subsequent Principal Component Analysis (PCA) with Promax rotation revealed 2 factors which were connected to (i) *prosodic and long term attributes* and to (ii) *segmental attributes*. Even though, the first dimension was linked to prosody it also comprises attribute scales that are specific to the voice of the systems, such as naturalness and pleasantness.

3.2. Test 2

In [6], a pilot study was conducted in order to unveil the perceptual quality dimensions of the Festival synthesizer [7]. 8 sentences from the TIMIT database [8] were chosen and synthesized with an English-speaking female voice. The stimulus duration varied from 1.9 to 4.1 seconds. 8 native speakers of English which were all experienced with listening to synthetic speech took part in a paired comparison (PC) test. They were instructed to rate whether the two presented stimuli were *similar* or *different* in terms of naturalness. The responses were compiled into a dissimilarity matrix which was then processed via an MDS analysis. The resulting dimensions were interpreted through visual and auditory analysis of the configuration of the stimulus space. The first dimension represents (i) *prosodic cues* which reflect the appropriateness of duration and intonation. The second dimension is linked to (ii) *segmental and unit-level cues*. It describes the appropriateness of units selected for synthesis as well as the number of selected units.

3.3. Test 3

To test the reliability and validity of the test method proposed in the ITU-T Rec. P.85 [9], Viswanathan et al. [2] conducted a series of 5 consecutive listening tests. In the final study, stimuli produced by 5 English-speaking TTS systems were evaluated on 9 5-point ACR scales. Additionally, participants were instructed to also rate the overall quality and the acceptability of the systems. The investigated systems used either phones or sub-phone units for concatenative synthesis. The synthesizers included algorithmic variations for pitch and duration generation. The stimuli were rated by 128 naïve test participants. A factor analysis revealed 2 factors: Dimension 1 is related to the extent to which speech is similar to natural human speech and was thus labeled (i) *naturalness*; Dimension 2 describes how well the content of the signal can be understood, hence it can be assigned to the (ii) *intelligibility* of the signal.

3.4. Test 4

In [10], speech material from 6 German "off-the-shelf" TTS systems was evaluated. The stimuli were created by diphone-based synthesizers using the pitch-synchronous-overlap-add (PSOLA) technique and unit-selection systems. A total of 10 speech samples have been generated per TTS system, half for male speakers and half for female ones. The synthesized speech samples have an average duration of 12s and consist of two utterances separated by a silence interval of approximately 2s. The listening test closely followed the ITU-T Rec. P.85 [9]. Thus, besides the rating of the stimuli on 8 ACR scales, the 17 test participants were also given a parallel task. As suggested in P.85, the listening test also included natural speech reference files. A subsequent Principal Axis Factor (PAF) analysis with Promax rotation revealed 2 dimensions. The first dimension consists of scales concerning the naturalness of the synthesized voice as well as prosodic attributes of the signal. The second dimension comprises scales that cover the fluency and intelligibility of the signal. Thus, dimension 1 was labeled (i) *naturalness and prosody* while dimension 2 was named (ii) *intelligibility*.

3.5. Test 5

In [11], Mayo et al. pursued the investigations described in Section 3.2. 24 sentences from the TIMIT corpus were selected and synthesized with an English-speaking female voice by the diphone-based Festival speech-synthesis system. The average duration of the stimuli was 2.7s. 30 participants took part in a PC test, where they were instructed to rate the similarity of a pair of stimuli in terms of naturalness. Two types of acoustic analysis were carried out: the automatic analysis consisted of measures that were computed by Festival during the synthesis process (e.g., target and join costs) and measures that were derived from those features (e.g., total cost, target costs of different types of diphones); the manual analysis included comparisons with natural speech files (e.g., number of transcription/pronunciation errors per synthetic utterance).
A subsequent MDS analysis yielded 3 dimensions. Through visual, auditory and cluster analysis these dimensions could be linked to (i) *overall join quality/quantity*, (ii) *join distribution and detectability*, and (iii) *unit appropriateness and prosody*. In our view the first two dimensions are thus connected to segmental attributes that concern the fluency and the intelligibility of the speech signal, while the third dimension represents global characteristics that describe the prosodic quality of the signal.

3.6. Test 6

Test 6 was part of an extensive study [12] in which the inherent quality dimensions of state-of-the-art TTS systems were investigated. 16 German-speaking synthesizers (formant synthesizers, PSOLA-based diphone synthesizers, unit-selection systems, and HMM-synthesizers) were used to generate 2 samples for each of the 30 different configurations[1] of synthesizers. The average duration was 10s. All stimuli were rated by 30 participants on 16 continuous scales (CS) that were developed dur-

[1] A configuration denotes a specific combination of one voice and one synthesis system

Table 1: Comparison of the main characteristics of the different test setups for Test 1-9.

	TEST 1	TEST 2	TEST 3	TEST 4	TEST 5	TEST 6	TEST 7	TEST 8	TEST 9
Year	1995	2005	2005	2007	2011	2011	2011	2012	2012
Language	German	English	English	German	English	German	German	German	English
Synthesizer type:									
Formant	✓					✓		✓	
Concatenative	✓		✓	✓		✓		✓	✓
Unit-selection		✓		✓	✓	✓	✓	✓	✓
HMM						✓		✓	✓
Number of systems	5	1	5	6	1	16	2	20	10
Number of configurations	5	1	5	12	1	30	5	57	10
Stimuli per configuration	6	8	9	5	24	2	8	1	22
Quality assessment via	ACR	PC	ACR	ACR	PC	CS	CS	CS	CS
Number of scales	8	-	9	7	-	16	9	16	7
Length of stimuli	100 words	1.9-4.1s	20-25 words	12s	2.7s	10s	55s	5s	45s

ing two extensive pretests. A subsequent PAF with Promax rotation revealed 3 perceptual dimensions. The first and most broad dimension comprises scales like accentuation, naturalness, rhythm, and pleasantness. Thus, it was labeled (i) *naturalness*. The second dimension consists of scales that specify (ii) *disturbances* in the signal, e.g., hissing and noise. The last dimension is related to (iii) *temporal distortions*, e.g., concatenation artifacts which occur in unit-selection synthesis. Additionally, the scale (iv) *speed* appeared to be a supplementary dimension.

3.7. Test 7

In [13], a pilot study was conducted to find a suitable set of attribute scales for the quality assessment of TTS in audiobook reading tasks. 2 German-speaking unit-selection synthesizers with female and male voices were used to synthesize book passages from 8 different books. The passages had an average duration of 55s and were chosen with the intention to cover a broad variety of different writing styles. Attribute scales from the P.85 questionnaire as well as scales that were developed especially for the evaluation of TTS-read audiobooks were used in this test. A PAF analysis with Promax rotation yielded 2 dimensions: the first dimension includes scales like voice pleasantness and listening effort and is thus related to the (i) *listening pleasure*; the second dimension comprises scales like intonation and speech pauses, hence it reflects the (ii) *prosody & rhythm* of the speech signal.

3.8. Test 8

This database was gathered during a study [3] which aimed to complement and to expand the results from Section 3.6. Therefore, 30 female and 27 male stimuli with an average duration of 5s were generated from the same utterance by different configurations of German-speaking TTS systems (formant synthesizers, PSOLA-based diphone synthesizers, unit-selection systems, and HMM-synthesizers). The stimuli were evaluated by 40 naïve test participants in a sorting task. The resulting dissimilarity matrix was processed via an MDS analysis and yielded 3 perceptual quality dimensions.
In a post-test, all stimuli were rated on the same 16 CS as described in Test 6. 12 test participants (5 expert listeners from the Quality and Usability Lab of the TU Berlin and 7 naïve subjects) took part in the test. Subsequently, the 3 quality dimensions were interpreted by means of expert listening and the ratings on the 16 attribute scales: dimension 1 describes voices with personality and charisma and was thus labeled (i) *naturalness of voice*; the second dimension is related to concatenation artifacts as well as the prosody of the signal, hence it is describes (ii) *temporal distortions*; the third dimension distinguishes between relaxed and slow speaking TTS systems and synthesizers which generate stressed and restless sounding voices, therefore it was labeled (iii) *calmness*.

3.9. Test 9

This database [14] was gathered within the scope of the TTS-audiobook-reading task of the Blizzard Challenge 2012 [15]. The results from the pilot study in Section 3.7 were the basis of the experimental setup. 10 male English-speaking synthesizers were used to synthesize book passages from 13 different books. The passages had an average duration of 45s. As in Test 7, the passages were selected with the aim to cover different writing styles. The recommendations from [13] lead to several changes in the selection and the labelling of the attribute scales. A PAF analysis with Promax rotation yielded 2 dimensions which mainly confirmed the dimensions (i) *listening pleasure* and (ii) *prosody and rhythm* from Test 7.

4. Similarities and differences

This section outlines the similarities and differences of the studies presented in the previous section and their impact on the resulting quality dimensions. An overview of all relevant characteristics of the experimental setup is discussed in the following and can be seen in Table 1.
As mentioned in Section 2, the quality-assessment method has a major influence on the resulting quality dimensions. The ratings in listening tests that use attribute scales to assess quality are always limited on the presented scales. Thus, characteristics that cannot be expressed by any of the scales are not captured. However, the experimental setups in Test 6 and 8, where 16 scales

Table 2: Perceptual quality dimensions of synthetic speech.

Dimensions	Relevant Scales	Test 1	Test 2	Test 3	Test 4	Test 5	Test 6	Test 7	Test 8	Test 9
Naturalness of Voice	*naturalness voice pleasantness*	Prosody		Naturalness	Naturalness and Prosody		Naturalness	Listening Pleasure	Naturalness of Voice	Listening Pleasure
Prosodic Quality	*stress rhythm prosody intonation*		Prosodic Cues	Intelligibility		Unit Appropriatness and Prosody		Prosody & Rhythm	Temporal Distortions	Prosody & Rhythm
Fluency and Intelligibility	*fluency intelligibility bumpiness polyphony*	Segmental	Segmental or Unit Level Cues		Intelligibility	Overall Join Quality/Quantity Join Distribution and Detectability	Temporal Distortions			
Absence of Disturbances	*hissing noise rasping disturbances*						Disturbances			
Calmness	*speed tension*						Speed		Calmness	

were presented, are more likely to give a deeper insight into the perceived quality. Nevertheless, one cannot be certain that naïve listeners which do not have detailed knowledge about the quality characteristics of speech, all understand the wording of the scales in the same way. In contrast, the PC test and the sorting task with subsequent MDS analysis bypass this constraint, but there is no information on the interpretation of the resulting stimulus space.
Moreover, the resulting quality dimensions also depend on the different types of synthesizers that were part of the test database. Thus, synthesizer-specific characteristics, e.g., the noise of HMM-synthesizers or the sonic glitches of concatenative systems, can naturally only be assessed if these types of systems are part of the study. Accordingly, studies that only feature formant synthesizers and diphone based concatenative systems, e.g., as in Test 1, are most likely to lead to different dimensions than studies that only assess unit-selection synthesizers, e.g., as in Test 2 and 5.
Furthermore, the duration of the generated stimuli also affects the perceived quality. The stimuli from the audiobook-reading tasks in Test 7 and 9, with durations of 55s and 45s, respectively, could bring other quality aspects into focus than stimuli from a different use case. In addition, very short stimuli, as in the Tests 2 and 5, can be difficult to judge in terms of voice or prosodic quality.

5. Results

The differences in the quality assessment methods, the synthesizer types used in the tests, and the different stimulus durations in most of the studies indicate ambiguous results. In the following, we present a comparative overview of the perceptual quality dimensions resulted from the studies in Section 3 and show that these dimensions can be linked to 5 universal perceptual quality dimensions of synthetic speech which are:

- **naturalness of voice**
- **prosodic quality**
- **fluency and intelligibility**
- **absence of disturbances**
- **calmness**

5.1. Naturalness of voice

As can be seen in Table 2, the dimension *naturalness of voice* is part of the outcome of most studies, with the exception of the MDS experiments (Test 2 and 5). However, this can be explained considering the stimuli from those tests: they were all generated from the same voice by the Festival synthesizer. Thus, none of them differed in voice characteristics. Even though the first dimension in the two audiobook tests was labeled *listening pleasure*, which seemed more suitable for this use case, it actually represents the character of the voice.

5.2. Prosodic quality

Due to the overlap of the first two dimensions in some studies (Test 1, 4, and 6) the second dimension seems to be more vague. Test 7 and 9, on the other hand, show that these dimensions can indeed be regarded as independent dimensions, even though they are somewhat correlated [13] [14]. The prosodic dimension can also be retrieved in Test 2 and 5, where the test participants did not perceive a dimension concerning the voice of the signal.

5.3. Fluency and intelligibility

The third prominent dimension covers *fluency and intelligibility* and it can be found in all studies except the audiobook experiments. This dimension captures segmental artifacts that are characteristic for synthesizers that concatenate smaller units like diphones. Considering the requirement for high-quality voices in audiobook reading tasks with very few glitches in concatenation, this dimension is indeed not prominent. The MDS study by Mayo from 2011 (Test 5) shows that this dimension can be further split up, at least for unit-selection synthesizers. On the contrary, the overlap of the prosody and the fluency/intelligibility dimensions in the MDS experiment (Test 8) shows that these two dimensions are hard to distinguish for naïve listeners.

5.4. Absence of disturbances

The dimension *absence of disturbances* could only be retrieved from the extensive experiments in Test 6. This is most likely due to the fact that the presented scales were developed with the help of speech and audio experts which might focus on various types of degradations. Even though the test participants could clearly distinguish, e.g., the grade of noise and hiss in the signal, these degradations were obviously less important to them than issues concerning the voice or the prosody of the signal. Nonetheless, this dimension can be useful to assess the quality of HMM synthesizers or systems that concatenate coded speech units which can produce noisy speech signals.

5.5. Calmness

Finally, the dimension *calmness* was found in Test 6 and 8. This dimension however appears to be less important since most of the speech synthesizers run at a similar speech rate. Nonetheless, when assessing the quality of fast synthesizers, like they are deployed in reading devices for the blind, this quality aspect can play a crucial role.

6. Conclusions and future work

Even though this study combined the results of 9 different experiments, further research will be needed to confirm the 5 resulting dimensions. Especially the relevance of the dimensions 4 and 5 should be further investigated. Nonetheless, including scales marked as relevant in Table 2 in future listening tests is expected to provide a more complete view on the perceptual aspects of the systems.

Furthermore, this study can be a basis for changes in the often criticized evaluation protocol P.85. Being developed in 1995, way before the era of high-quality synthesizers of today, this recommendation could be revised considering the results from this study.

As a concluding remark, we can say that the three major quality dimensions of synthetic speech are (i) *naturalness of voice*, (ii) *prosodic quality*, and (iii) *fluency and intelligibility*. Thus, even though the intelligibility of TTS systems substantially increased over the past decade, this dimension is still important for the perceptual quality.

7. Acknowledgements

The present study was carried out at Quality and Usability Lab, TU Berlin. It was supported by the Deutsche Forschungsgemeinschaft (DFG), grants MO 1038/11-1 and HE 4465/4-1.

8. References

[1] V. Kraft and T. Portele, "Quality Evaluation of Five German Speech Synthesis Systems," *Acta Acustica 3*, pp. 351–365, 1995.

[2] M. Viswanathan and M. Viswanathan, "Measuring Speech Quality for Text-to-Speech Systems: Development and Assessment of a Modified Mean Opinion Score (MOS) Scale," *Computer Speech and Language*, vol. 19, pp. 55–83, 2005.

[3] F. Hinterleitner, C. Norrenbrock, S. Möller, and U. Heute, "What Makes this Voice Sound so Bad? A Multidimensional Analysis of State-of-the-Art Text-to-Speech Systems," *Proc. of the 2012 IEEE Workshop on Spoken Language Technology (SLT 2012)*, pp. 240–245, 2012.

[4] I. Borg and G. P., *Modern Multidimensional Scaling - Theory and Applications, 2nd edition*. Springer Series in Statistics, New York, 2005.

[5] L. Tsogo, M. Masson, and A. Bardot, "Multidimensional Scaling Methods for Many-Objects Sets: A Review," in *Multivariate Behavioral Research*, vol. 35, no. 3, 2000, pp. 307–319.

[6] C. Mayo, R. A. J. Clark, and S. King, "Multidimensional Scaling of Listener Responses to Synthetic Speech," *Proc. of the 6th Annual Conference of the International Speech Communication Association (Interspeech)*, pp. 1725–1728, 2005.

[7] "The Festival Speech Synthesis System." [Online]. Available: http://www.cstr.ed.ac.uk/projects/festival/

[8] J. S. Garofolo, *Getting Started with the DARPA TIMIT CD-ROM: An Acoustic Phonteic Continuous Speech Database*, National Institute of Standards and Technology (NIST), Gaithersburgh, MD, 1988.

[9] ITU-T Rec. P.85, *A Method for Subjective Performance Assessment of the Quality of Speech Voice Output Devices*, International Telecommunication Union, Geneva, 1994.

[10] K. Seget, *Untersuchungen zur auditiven Qualität von Sprachsyntheseverfahren (Study of Perceptual Quality of Text-to-Speech Systems)*. Diplomarbeit, Lehrstuhl für Netzwerk- und Systemtheorie, Christian-Albrechts-Universität Kiel, 2007.

[11] C. Mayo, R. A. J. Clark, and S. King, "Listeners' Weighting of Acoustic Cues to Synthetic Speech Naturalness: A Multidimensional Scaling Analysis," *Speech Communication*, vol. 53, pp. 311–326, 2011.

[12] F. Hinterleitner, S. Möller, C. Norrenbrock, and U. Heute, "Perceptual Quality Dimensions of Text-to-Speech Systems," *Proc. of the 12th Annual Conference of the International Speech Communication Association (Interspeech 2011)*, pp. 2177–2180, 2011.

[13] F. Hinterleitner, G. Neitzel, S. Möller, and C. Norrenbrock, "An Evaluation Protocol for the Subjective Assessment of Text-to-Speech in Audiobook Reading Tasks," in *Proceedings of the Blizzard Challenge Workshop. International Speech Communication Association (ISCA)*, Turin, Italy, 2011.

[14] F. Hinterleitner, C. Norrenbrock, and S. Möller, "Perceptual Quality Dimensions of Text-To-Speech Systems in Audiobook Reading Tasks," *Proc. of the 24th Konferenz Elektronische Sprachsignalverarbeitung (ESSV), Bielefeld, (Germany)*, 2013.

[15] "The Blizzard Challenge 2012." [Online]. Available: http://www.synsig.org/index.php/Blizzard_Challenge_2012

Evaluation of contextual descriptors for HMM-based speech synthesis in French

Sébastien Le Maguer[1], *Nelly Barbot*[1], *Olivier Boeffard*[1]

[1]IRISA - University of Rennes 1, Lannion, France

{Sebastien.Le_Maguer, Nelly.Barbot, Olivier.Boeffard}@irisa.fr

Abstract

In HTS, a HMM-based speech synthesis system, about fifty contextual factors are introduced to label a segment to synthesize English utterances. Published studies indicate that most of them are used for clustering the prosodic component of speech. Nevertheless, the influence of all these factors on modeling is still unclear for French.

The work presented in this paper deals with the analysis of contextual factors on acoustic parameters modeling in the context of a French synthesis purpose. Two objective and one subjective methodologies of evaluation are carried out to conduct this study. The first one relies on a GMM-approach to achieve a global evaluation of the synthetic acoustic space. The second one is based on a pairwise distance determined according to the acoustic parameter evaluated. Finally, a subjective evaluation is conducted to complete this study.

Experimental results show that using phonetic context improves the overall spectrum and duration modeling and using syllable informations improves the F0 modeling. However other contextual factors do not significantly improve the quality of the HTS models.

Index Terms: HTS, Evaluation, Contextual factors, French synthesis

1. Introduction

Based on Hidden Markov Models, HTS [1] provides a framework to synthesize speech using parametric statistical models offering a good flexibility. Acoustical parametrization is generally done with a MLSA filter [2] associated with the STRAIGHT model [3]. To produce an acoustic signal for a specific utterance, the temporal evolution of the acoustic parameters is generated from a sentence-level HMM whose observations encompass segmental and prosodic (f0 and duration) informations. This sentence-level HMM is built by concatenating HMM related to the phonemes which compose the utterance.

In HTS, a phone is qualified by a set of contextual factors. For example, the set of describing factors for English, introduced in [4], contains about fifty contextual descriptors associated to the phonetic, phonologic, prosodic or linguistic properties of a segment. During the HTS clustering stage, these factors are used to guide the construction of a decision tree. Consequently, contextual factors have an important role in model training which implies that choosing a proper set could influence the quality of the contextual HMM.

Even though HMM-based synthesis systems are evaluated during the Blizzard challenge [5], only few studies are focused on the influence of contextual factors on the acoustical parameter modeling and, then, on the synthesis achieved by HTS. In [6], the acceleration parameters are studied by computing distances between generated parameters containing acceleration coefficients and those without acceleration coefficients. This study shows that discarding acceleration coefficients implies a saw-tooth trajectory generation. In [7], the duration prediction error is evaluated using RMSE and the correlation between the generated duration and the original duration associated to the same utterance. The results in [7] indicate differences in modeling the duration of consonants and vowels. Specifically to the contextual feature issue, the contribution of high level linguistic features along with the influence of hand labeled versus automatic labeled features are assessed in [8]. This study shows that using automatic annotation in the training labels could affect HTS modeling and that pitch accents, boundary tones and POS (Part-Of-Speech) tags contribute more than other phrase level contextual features to the modeling. By extending this result, we can assume that some contextual features are less effective than others. This assumption is confirmed by the prosodic contextual factor evaluation conducted by [9] to identify a minimal descriptive feature set. Finally, using this assumption, it is possible to achieve "on-the-fly" synthesis like the one proposed in [10].

The aim of this paper is to propose a protocol for an objective evaluation of the HTS synthesis and to apply this protocol to analyze the speech generated by HTS for French. The first method we propose is based on an acoustic space modeling. By analogy to voice conversion, we assume that the acoustic space is well represented by a Gaussian Mixture Model (GMM). By comparing the likelihood of each GMM, which models a generated acoustic space, given a reference speech dataset, it is possible to compare the similarity of the different acoustic spaces. In this way, we can study the quality of the acoustic spaces generated according to different sets of contextual features. In addition, comparing acoustic spaces using a GMM likelihood does not require an alignment between the HTS synthetic speech and the natural reference. Then this approach enables an evaluation of acoustic parameters independently of the duration. However, we need enough data to train the GMM which prevents precise analysis by using this method. Consequently, a second objective methodology, based on usual distances, is used in this protocol for local analysis. During experiments these distances allow to assess the modeling quality according to phonetic categories. In addition, a global subjective MOS test is proposed to compare synthesized speech obtained with different contextual factor combinations and natural speech.

This paper is organized as follows. Section 2 presents the objective evaluation protocol. Section 3 exposes data and the results of the experiments. Section 4 describes the subjective evaluation protocol and its results.

2. Objective evaluation

2.1. General framework

The purpose of the proposed protocol is to study the influence of various descriptors on the acoustic space generated from a single-speaker HTS system, and to assess its proximity to the acoustic space associated with natural speech of the same speaker.

The set of descriptors used to qualify a phonetic segment is the one given by [4] with some adaptations. First, information concerning lexical accent at the syllable level and the TOBI labels at the sentence level are overlooked. Secondly, we used specific French tools to retrieve the POS tags. The descriptors are introduced in table 1. In order to achieve our study, several subsets of contextual factors have been defined. They are presented in table 2.

The acoustic space of the speaker, estimated from STRAIGHT analysis-by-synthesis signals, will serve as a reference. In this specific case, denoted a/s, the HTS system is not used (it is the best case for the objective evaluation experiments). In the following paragraphs, the notations $L_{a/s}$, $V_{a/s}$ and $T_{a/s}$ will refer to three sets of acoustic vectors (corresponding respectively to the Learning, Validation and Test corpora) stemming from analysis-by-synthesis signals, corresponding to disjoint sets of utterances.

For each subset of contextual factors $k \in \{$p1,...,p5-s_pos$\}$, the learning phase of the HTS is done on the $L_{a/s}$ corpus using the k set only. A corpus L_k of acoustic vectors (respectively V_k and T_k) is generated by HTS, corresponding to the same utterances as $L_{a/s}$ (respectively $V_{a/s}$ and $T_{a/s}$).

In order to compare the acoustic spaces generated by HTS with the one based on analysis-by-synthesis signals, two objective evaluation methods are being considered. One is based on a GMM modeling of the acoustic spaces, and the other relies on a distance between the vectors generated by HTS and the vectors stemming from analysis-by-synthesis processing. In order to assess, in an independent way, the quality of each HTS parameter, the duration of the segments observed in $T_{a/s}$ is forced upon the generation process of the elements of T_k (for each $k \neq a/s$ set) in case of the evaluation of MGC (Mel Generalized Cepstral) coefficients and F0 values synthesized by HTS.

2.2. Evaluation based on GMM

This methodology mainly relies on the following assumption: if a configuration of HTS improves the quality of the synthesized speech signal, the likelihood of the reference data with respect to the acoustic space generated by HTS should increase. Since the likelihood depends on both the model and the data, we have chosen to keep the same test corpus $T_{a/s}$ as a referential throughout this evaluation process.

For every $k \in \{$a/s,...,p5-s_pos$\}$, the GMM $\mathcal{M}_k$ is learnt over L_k using an EM algorithm. According to the evaluated HTS parameters, each vector of L_k could correspond to the spectral part of frames, the fundamental frequency of frames or the duration of phones. In case of evaluation of the spectral part, L_k vectors are 39-order MGC coefficient vectors. The 0^{th} MGC coefficient corresponds to the gain and is ignored in order to facilitate the comparison with the evaluation based on mel-cepstral distortion (eq.1).

For each of these data types, the GMM-based evaluation methodology is similar. However, in case of the spectral evaluation, a principal component analysis (PCA) is operated on the whole set of learning corpora in order to reduce the dimension of their elements and ensure the numerical stability during the learning stage of $\mathcal{M}_k$. The target threshold of the PCA is at least 95% of the explained variance in the data. The PCA linear transformation $\mathcal{T}$ is also applied to the vectors of V_k, T_k and $T_{a/s}$ so as to homogenize the data. After the application of the PCA, with no risk of confusion, notations L_k, V_k and T_k are conserved.

The number of components n of the GMM $\mathcal{M}_k(n)$ is determined using the validation corpus V_k: for $i \in [1..9]$, the $\mathcal{M}_k(n)$ model is learnt over L_k for $n = 2^i$ and the log-likelihoods $LL(L_k|\mathcal{M}_k(n))$ and $LL(V_k|\mathcal{M}_k(n))$ are then computed. The covariance matrices of the GMM components are diagonal. Finally, an over-learning situation is detected when $LL(V_k|\mathcal{M}_k(n)) << LL(L_k|\mathcal{M}_k(n))$. The optimal value of n, known as $n^\star$, is then chosen as the minimal number 2^i so that $LL(L_k|\mathcal{M}_k(n)) - LL(V_k|\mathcal{M}_k(n)) \geq \epsilon$, for every $k \in \{a/s, \ldots, p5 - s\_pos\}$, where ϵ was a priori defined to $\epsilon = 0.2$.

The log-likelihoods of the data from the test corpora $LL(T_{a/s}|\mathcal{M}_k(n^\star))$ and $LL(T_k|\mathcal{M}_k(n^\star))$ are then computed, along with the associated 95% confidence intervals using a Bootstrap methodology. This allows for the evaluation of the adequacy of the HTS-generated acoustic spaces with the reference data coming from the analysis-by-synthesis STRAIGHT process.

2.3. Evaluation based on distances

The aim of this methodology is to dispose of a measure that enables a local analysis of the closeness between the coefficients generated by HTS and those stemming from the STRAIGHT analysis.

In case of the evaluation of MGC vectors and F0 values, the segments provided by HTS have the same duration as the STRAIGHT segments and the frames of T_k and $T_{a/s}$ can be matched for each $k \in \{$p1,...,p5-s_pos$\}$.

The measure considered here between two 39-order MGC vectors c_k and $c_{a/s}$, respectively elements of T_k and $T_{a/s}$, is the mel-cepstral distortion expressed as

$$D\left(c_k, c_{a/s}\right) = \frac{10\sqrt{2}}{\ln(10)} \sqrt{\sum_{i=1}^{39} \left(c_k(i) - c_{a/s}(i)\right)^2}. \quad (1)$$

This distortion is computed for all the $(c_k, c_{a/s})$ pairs of $T_k \times T_{a/s}$ and a confidence interval of the associated mean value is also computed for each $k \neq a/s$.

The distance between the F0 and duration values generated by HTS and their matched elements in $T_{a/s}$ is derived using a Root Mean Square error (RMS). More precisely, for each k subset, the RMS error between the F0 frames of T_k and $T_{a/s}$ is in cent. We have used 87 Hz as the reference frequency which represents the mean F0 value of the speaker. For the phone duration, the RMS error is computed taking into account all the phone instances of $T_{a/s}$.

At last, the voicing error rate has been introduced to complete the analysis of the fundamental frequency. This measure is used to analyze specifically the voicing boundary F0 modeling. Considering the F0 values c_k and $c_{a/s}$, respectively elements of T_k and $T_{a/s}$, the voicing error is defined by

$$D\left(c_k, c_{a/s}\right) = \begin{cases} 0, & \text{if } c_k = c_{a/s} = 0 \\ 0, & \text{if } c_k \neq 0 \text{ and } c_{a/s} \neq 0 \\ 1, & \text{otherwise.} \end{cases} \quad (2)$$

	Id.	Horizon	Meaning
Pho.	A		Identity of the current segment
	B		Identity of the previous/next segment
	C		Identity of the previous-previous/next-next segment
Syllable	D	P/C/N	Number of phones + position of the current phone in the syllable
	E	C	Position of the syllable in the word
	F	C	Position of the syllable in the sentence
	G	P/C/N	Accented flag
	H	C	Number of syllables from the (previous accented)/current syllable to the current/(next accented) syllable
	I	C	Number of accented syllable before/after the current syllable in the sentence
	J	C	Vowel of the syllable
Word	K	P/C/N	Number of syllables in the word
	L	C	Position of the word in the sentence
	M	P/C/N	Word POS tag
	N	C	Number of words from the (previous content)/current word to the current/(next content) word
	O	C	Number of content words before/after the current word in the sentence
Sent.	P	P/C/N	Number of syllables in the sentence
	Q	P/C/N	Number of words in the sentence
	R	C	Position of the sentence in the utterance

Table 1: Contextual factors used for French speech synthesis. The second column indicates which items are described (P=Previous, C=Current and N=Next)

		A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
Seg.	p1	X																	
	p3	X	X																
	p5	X	X	X															
Syl.	p5-sy_pos	X	X	X	X	X	X												
	p5-sy_accent	X	X	X				X	X	X	X								
	p5-sy_full	X	X	X	X	X	X	X	X	X	X								
Word	p5-w_pos	X	X	X	X	X	X	X	X	X	X	X	X						
	p5-w_content	X	X	X	X	X	X	X	X	X	X			X	X	X			
	p5-w_full	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X			
Sentence	p5-s_pos	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

Table 2: Evaluated contextual factor sets. An "X" indicates that the factor belongs to the set.

The voicing error rate corresponds to the average voicing error value multiplied by 100.

3. Experiments and results

3.1. Data

The data used for the evaluation are extracted, using a full automatic process presented in [11], from an audiobook in French. The speaker was a male speaker whose reading was moderately expressive. The signal was sampled at 16kHz. The HTS version is the speaker-dependent architecture presented at the Blizzard challenge in 2005[1]. Utterances from $L_{a/s}$ are used to train the HMM models in HTS for each contextual descriptor set k under consideration.

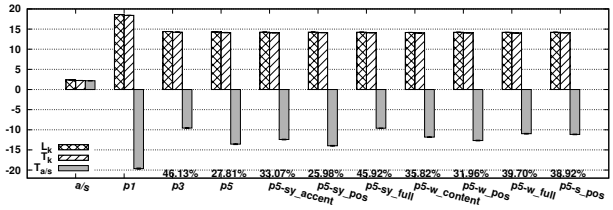
As previously mentioned, three sets of utterances are built: a training corpus containing about 300 utterances for a duration of one hour, a test corpus and a validation corpus which both contain 152 utterances for a duration of 10 minutes. Furthermore, for the two objective evaluations, all frames associated with non speech sound labels (pauses, noises, etc) are simply discarded. Therefore, the training corpus contains about 520,000 frames; the test and validation corpora contain about 85,000 frames each.

3.2. GMM-based evaluation results

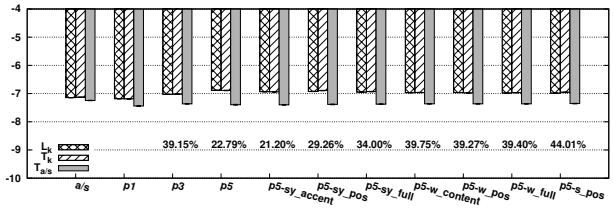
At the end of the GMM learning stage described in section 2.2, the resulting GMM are composed of 512 Gaussians for the spectral part evaluation, 128 for the F0 and 2 for the duration. Furthermore, for the spectral part evaluation, the application of the PCA reduces the data dimension from 39 to 12. Results of the GMM evaluation method are illustrated in figure 1.

For all evaluated acoustic parameters and considering $T_{a/s}$ as a reference, the highest likelihood of its elements is obviously provided by $\mathcal{M}_{a/s}$ and the lowest one by $\mathcal{M}_{p1}$: using only the phonetic label of the current phone segment is not enough to generate an appropriate acoustic space according to the coefficients extracted from the natural speech signal. Globally, by taking into account the closest phonetic context (one left and right phonetic context), the likelihood of the data stemming from analysis-by-synthesis and relative to the GMM generated from HTS acoustic vectors increases significantly.

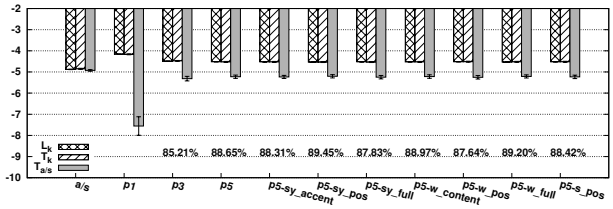
Differences between acoustic parameters appear when more contextual factors are used. As for the segmental part, the best contextual factor set is p3 and, according to the presented results, taking into account more features sometimes could lead to produce more irrelevant data. As for the duration, we can observe that there is a constant improvement until the syllable level. However, concerning the duration, confidence intervals



(a) MGC avoiding the gain dimension



(b) F0



(c) Duration

Figure 1: Log-likelihoods of L_k, T_k and $T_{a/s}$ for GMM $\mathcal{M}_k$, where descriptor combination k is given in the x-axis. Rates below the bars indicate the improvement rates $(LL(T_{a/s}|\mathcal{M}_k) - LL(T_{a/s}|\mathcal{M}_{p1}))/(LL(T_{a/s}|\mathcal{M}_{a/s}) - LL(T_{a/s}|\mathcal{M}_{p1}))$ associated to each k from $p3$ to $p5 - s\_pos$ compared to $p1$.

overlap which means that from p3 to p5-s_pos, all contextual factor sets are equivalent. Finally, results achieved by the evaluation for F0 indicate that all contextual factor sets are equivalent. So, HTS globally provides suitable F0 space with respect to the analysis-by-synthesis data.

3.3. Pairwise evaluation results

3.3.1. Spectral evaluation results

We present results of the second objective methodology based on mel-cepstral distortion between the original spectral coefficients and the ones generated by HTS, are illustrated in figure 2.

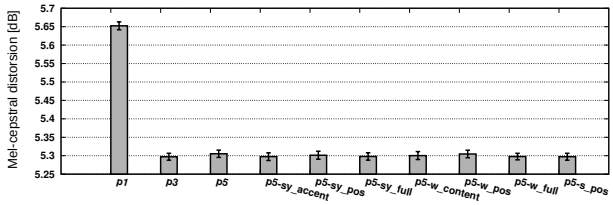


Figure 2: Average mel-cepstral distortion between $T_{a/s}$ and T_k vectors for several combinations of contextual descriptors presented on the x-axis

These results show that speech generated using only the current phonetic label of a segment (p1 set) is farthest from the natural speech signal. This is consistent with the GMM evaluation results. In addition, the lowest distortion is achieved using the set p3 with no significant difference with more complete contextual factor sets. Furthermore, according to the results provided by the GMM based evaluation, this can mean that using some sets, like p5 for example, leads to consider some analysis-by-synthesis values unlikely even if the generated values are not so far from them.

In order to post-analyze potential sources of errors, sets of vectors are defined according to the phonetic characteristics (consonant/vowel, voiced/unvoiced/oral/nasal, etc) and the associated mel-cepstral distortions are given in figure 3.

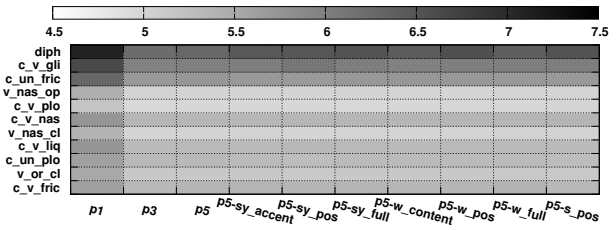


Figure 3: Mel-cepstral distortion presented by phonetic characteristics for each contextual factor set. Distortion, expressed in dB, is quantified on a grayscale. Each set is labeled by x_y_z where x$\in$ {c(onsonant), v(owel)}, y$\in$ {v(oiced), un(voiced), or(al), nas(al)} et z$\in$ {gli(de), fric(ative), (im)plo(sive), op(en), cl(osed), liq(uid), nas(al)}. Diph. set is represented only by the phone /yi/.

The distortion values between analysis-by-synthesis coefficients and generated coefficients based on p1 descriptor set are the highest ones. Confidence intervals, which are not present in this figure, confirm that those differences are significant relatively to other contextual descriptor combinations. By comparing mel-cepstral distortions between the different descriptor sets, we distinguish three main sets: vowels with voiced plosives, diphthongs and unvoiced fricatives, and the other consonants. None of contextual factors introduced in the French set seem to fill the gaps between those main sets. Considering the diphthong, the distortion can be explained by number of frames (about 2.000 frames) used in the training stage but this explanation is not suitable for other phonetic sets (from 7 to 90 times greater). We conclude that none of the contextual descriptors used can really capture the specific acoustic properties of, for example, glides as much as open nasal vowels.

3.3.2. F0 evaluation results

The results obtained by applying the pairwise evaluation on the F0 are presented in figures 4 and 5. Using high-level contextual factors does not improve the error rate. Indeed, the best voicing error rate is achieved by using the direct phonetic context (labels of the previous, current and next segments). However, by comparing the RMS values, we notice a constant improvement until the set p5-sy_full. Taking into account higher level contextual factors implies a statistically significant improvement of the RMS. So, according to those results, the best contextual factor set for the F0 modeling is p5-sy_full.

By comparing these results with the GMM-based evaluation ones, we can notice a clear difference. If we analyze globally the generated values, most of contextual factor sets lead to produce equivalent F0 spaces. However differences between the generated F0 values occur more locally. So, even if the F0 values produced by most of the contextual factor sets are consistent, p5-sy_full leads to generate the closest F0 values to the

analysis-by-synthesis ones.

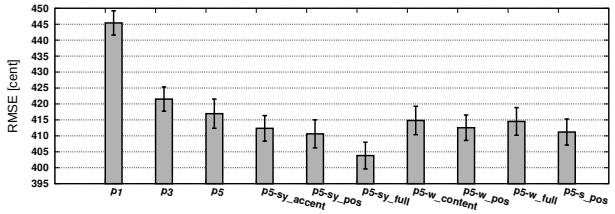


Figure 4: Global RMS error for the F0 component

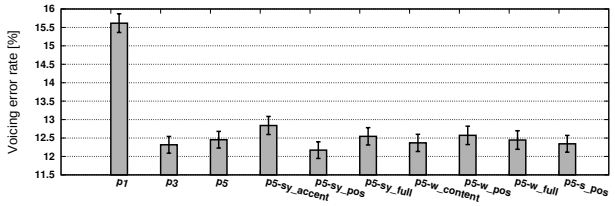


Figure 5: Global voicing error rate

In order to complete this analysis, we compute the RMS error in cent and the voicing error rate for each category of phones. The results are, respectively, presented in figures 6 and 7. In both cases two trends stand out. By comparing the contextual factor sets, we find that the improvement achieved by p3 relatively to p1 can be explained by specific categories like the voiced plosive (RMSE) or the voiced liquid (voicing error rate). Considering the diphthong, we cannot conclude as the number of frames is low comparing to other phonetic categories.

As we just mentioned, differences in the quality of the F0 modeling between phonetic categories can be observed. Unvoiced plosive and unvoiced fricative modelings are clearly worse than the others. This statement is valid in both measures. However, the voicing error rates associated with voiced segments are below 5%. Generally, the boundary of unvoiced labeled segment corresponds to a voicing boundary. Using MSD [12] implies that, during the training stage, one frame contributes to the voiced or the unvoiced distribution. This leads to a strict voiced/unvoiced split which implies problems at voicing boundaries. These results confirm that problem and indicate that no contextual factor set cannot avoid it even if using some specific factors could reduce this problem.

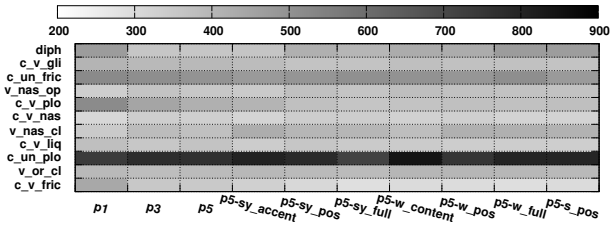


Figure 6: RMS error results by phoneme categories

3.3.3. Duration evaluation results

By applying the pairwise evaluation on the duration, we achieve results presented in figure 8. By comparing RMS error according to the contextual factor sets, the only significant difference

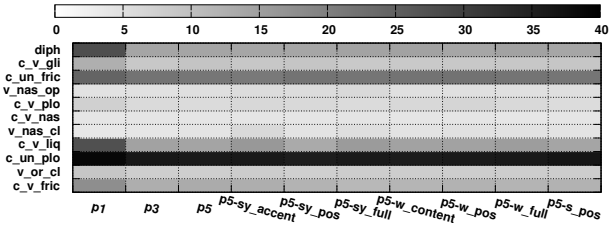


Figure 7: Voicing error results by phoneme categories

is provided by the p5 set in comparison of the p1 set. By comparison with GMM based evaluation, p3 results are more intermediate than the pairwise evaluation. These results indicate that the produced duration using p1 is not so distant than the ones provided using other descriptive features.

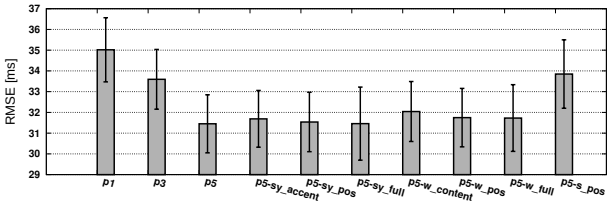


Figure 8: Global duration RMS error results

Finally, we focus the analysis by computing the RMSE for each phoneme. Results are presented in figure 9. The modeling of the phone /oe/ duration seems to be worse than other phones. However the confidence intervals, not detailed here, show that the difference is not statistically significant.

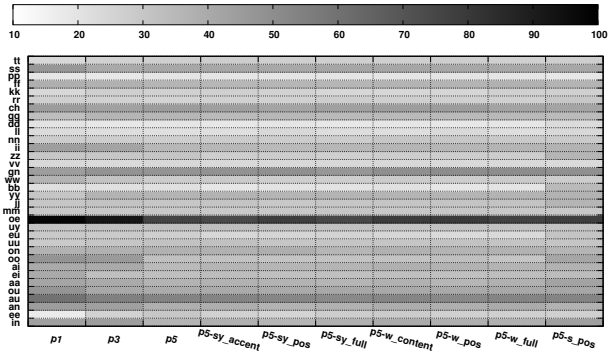


Figure 9: Duration RMS error results by phoneme

3.4. Objective evaluation conclusions

According to all results achieved by objective evaluations, the more suitable contextual factor set is p5-sy_full. For all evaluations, a clear improvement is introduced by taking into account the direct phonetic context (p3 relatively to p1). However, although phonetic features suffice to achieve a fine prediction of the segmental part (the best set is p3) and the duration (the best set is p5), pairwise evaluation indicates that the modeling of the fundamental frequency requires more contextual factors. These results are consistent with studies achieved for other languages like the one given in [9]. Furthermore, our results also indicate that differences in the modeling quality exist between phonetic

categories. This statement is obvious in case of the F0 modeling. Actually, based on the study presented in [13], we assume that these differences could be due to the use of the MSD in the standard version of HTS and do not depend on a contextual factor set.

4. Subjective evaluation

4.1. Evaluation procedure

A global subjective evaluation was conducted in order to complete the analysis of the objective evaluation results.

In this evaluation, seven signal sets are defined : the natural signal, the analysis-by-synthesis signal and the signals produced by HTS according to five contextual factor sets. The three first sets are p1, p3 and p5. They are used to assess the impact of the phonetic context in the synthesis. The last two sets are p5-sy_full, which objective evaluations tend to indicate that it is the more suitable, and p5-s_pos which is the most complete contextual factor set. Each signal set contains thirty utterances extracted from the test corpus and the average duration of each signal is about six seconds.

The goal of this test is to evaluate the overall quality of the synthesis using a MOS score. Nine listeners, working in speech processing, have done this test. One hundred stimuli have been presented to each listener. So, the evaluation test for each listener has been about thirty minutes.

4.2. Subjective evaluation results

The subjective evaluation results are detailed in figure 10.

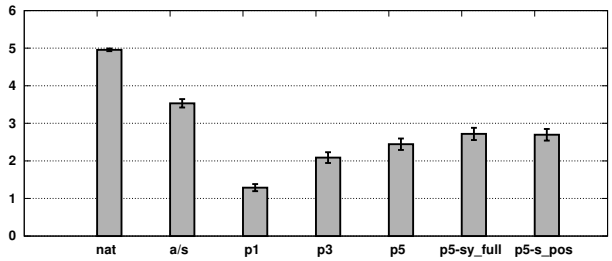


Figure 10: Results of the MOS evaluation

By comparing analysis-by-synthesis score against the score of the natural signal, we can notice that the listeners have perceived a signal damaging due to the signal parametrization. Among HTS synthesized signals, we can distinguish three sets: the signal synthesized using the combination p1, which has the lowest score; the signal synthesized using p3 whose quality is significantly improved against the signal p1 and the signals of the last three contextual factor combinations which have the highest score. However, signal deterioration due to the modeling is perceived, since all HTS synthesized signals are considered lower quality than the analysis-by-synthesis one.

As a significant improvement on the signal is perceived between p3 and p1, we assume that the modeling of each acoustic parameter is improved by taking into account the direct phonetic context. As for p5 and p5-sy_full, we assume that a better quality of the fundamental frequency modeling, done by HTS, is achieved by using syllable informations in the contextual factor sets. However, the subjective evaluation also confirms the results of the objective evaluations since no improvement was perceived between p5-sy_full and p5-s_pos.

5. Conclusion

In this paper, we have proposed an experimental protocol to objectively evaluate the synthesis achieved by HTS. This protocol is based on two complementary methods. The first one uses a GMM to model generated coefficient space and enables to assess the likelihood of the reference data according to this space. The second method relies on pairwise distances in order to carry out a more detailed analysis of the modeling achieved by HTS.

Using this protocol, we analyzed the closeness between the coefficients generated by HTS and those stemming from the STRAIGHT analysis for French synthesis. Experimental results suggest that using other descriptors than the phonetic and syllable context may be useless to improve the modeling achieved by HTS for this corpus. Based on the current methodology, further work must be achieved to analyze deeply the modeling achieved by HTS.

6. References

[1] H. Zen and T. Toda, “An overview of Nitech HMM-based speech synthesis system for blizzard challenge 2005”, Eurospeech, pp1957-1960, 2005.

[2] T. Fukada, K. Tokuda, T. Kobayashi and S. Imai,“An adaptive algorithm for mel-cepstral analysis of speech”, ICASSP, pp137-140, 1992.

[3] H. Kawahara, I.Masuda-katsuse and A. De Cheveigné, “Restructuring speech representations using a pitch-adaptive time frequency smoothing and an instantaneous-frequency-based F0 extraction: Possible role of a repetitive structure in sounds”, Speech Communication, 27, pp187-207, 1999.

[4] Tokuda, K., Zen, H. and Black, A. W., “An HMM-based speech synthesis system applied to english”, ICASSP, pp227-230, 2002.

[5] King, S. and Karaiskos, V., “The Blizzard challenge 2010”.

[6] Chen, Y.-n., Yan, Z.-j. and Soong, F. K., “A perceptual study of acceleration parameters in HMM-based TTS”, Interspeech, 2010.

[7] Silén, H., Helander, E., Nurminen, J. and Gabbouj, M., “Analysis of duration prediction accuracy in HMM-based speech synthesis”, Speech Prosody, 2010.

[8] Watts O., Yamagishi, J. and King, S., “The role of higher-level linguistic features in HMM-based speech synthesis”, Interspeech, 2010.

[9] Yokomizo, S., Nose, T. and Kobayashi, T., “Evaluation of prosodic contextual factors for HMM-based speech synthesis”, Interspeech, pp430-433, 2010.

[10] Astrinaki, M., d’Alessandro, N., Picart, B., Drugman, T. and Dutoit, T., “Reactive and continuous control of HMM-based speech synthesis”, SLT, pp252-257, 2012.

[11] Boeffard, O., Charonnat, L., Le Maguer, S., Lolive, D. and Vidal, G., “Towards fully automatic annotation of audiobooks for TTS”, LREC, 2012.

[12] Tokuda, K., Masuko, T., Miyazaki, N. and Kobayashi, T., “Hidden Markov models based on multi-space probability distribution for pitch pattern modeling”, ICASSP, 1999.

[13] Yu, K., Toda, T., Gasic, M., Keizer, S., Mairesse, F., Thomson, B. and Young, S., “Probablistic modelling of f0 in unvoiced regions in hmm based speech synthesis”, ICASSP, 2009.

Towards Speaking Style Transplantation in Speech Synthesis

Jaime Lorenzo-Trueba[1], *Roberto Barra-Chicote*[1], *Junichi Yamagishi*[2], *Oliver Watts*[2], *Juan M. Montero*[1]

[1]Speech Technology Group, ETSI Telecomunicacion, Universidad Politecnica de Madrid, Spain
[2]CSTR, University of Edinburgh, United Kingdom
{jaime.lorenzo, barra}@die.upm.es

Abstract

One of the biggest challenges in speech synthesis is the production of naturally sounding synthetic voices. This means that the resulting voice must be not only of high enough quality but also that it must be able to capture the natural expressiveness imbued in human speech. This paper focus on solving the expressiveness problem by proposing a set of different techniques that could be used for extrapolating the expressiveness of proven high quality speaking style models into neutral speakers in HMM-based synthesis. As an additional advantage, the proposed techniques are based on adaptation approaches, which means that they can be used with little training data (around 15 minutes of training data are used in each style for this paper). For the final implementation, a set of 4 speaking styles were considered: news broadcasts, live sports commentary, interviews and parliamentary speech. Finally, the implementation of the 5 techniques were tested through a perceptual evaluation that proves that the deviations between neutral and speaking style average models can be learned and used to imbue expressiveness into target neutral speakers as intended.

Index Terms: expressive speech synthesis, speaking styles, adaptation, expressiveness transplantation

1. Introduction

Speech synthesis is a field that has been seeing much more use in the last decade with the advent of human-machine interfaces, playing an integral role in them. As such there have been constant studies on how to improve its quality, naturalness, expressiveness, etc. Among these efforts is the project under which this investigation is enclosed: Simple4All [1]. Simple4All is an European funded project whose main purpose is to streamline the training process of expressive synthetic voices by creating a system that requires little to no supervision and is capable of learning constantly just by its interactions with the users.

More concretely, expressive speech synthesis is a sub-field of speech synthesis that has been drawing a lot of attention lately, as until recently there was no effort paid to increasing the adequacy of the produced voices to the task they were intended to be used in. But, if one were to assign expressiveness to the synthetic voices (e.g. emotions or speaking styles), the result would be a much more natural voice increasing the overall satisfaction of the end users of the interface. This, when considering the two main speech synthesis techniques (unit selection and HMM-based) places a serious restriction that clearly favors HMM-based synthesis [2]: if expressive data were to be recorded for every possible situation, the size of the databases would become immense, making unit-selection nonviable on principle. HMM-based synthesis, on the other hand, due to its parametric nature is much more adaptable, a fact that can be exploited even further by using adaptation techniques [3].

Consequently this study focuses on HMM-based synthesis and adaptation techniques in order to produce voices with the desired speaking styles. Firstly, and keeping in mind that the final training system should require minimal interaction from the user, it is interesting to minimize the training data required to produce the output models without reducing the final quality. This can be done by exploiting background average models [4] from which the final voice is adapted using one of the different available techniques (this study relies on CSMAPLR adaptation [3]). At this point the problem becomes how to imbue the models with speaking styles, towards which we can see some recent studies such as Cluster Adaptive Training [5], that relies on clustering the expressive training speakers into a continuous expressive speech space of the different available speaking styles.

The approach suggested in this paper consists of creating representative models for every desired expression from small subsets of data that clearly show the nuances of that particular oratory, including one for neutral or read-speech voices. Then we propose a way of modeling the differences between the neutral model and the target speaking style through adaptation transformations, together with a way of transferring this differences into a new neutral target speaker in order to adapt the voice into the desired speaking style pattern. This in the end allows us to generate voices with speaking styles for any target neutral speaker even if there is no previous expressive data available for that speaker. This technique is finally verified through a perceptual test, showing the usefulness of extrapolating the speaking style of average models as the results are considered by the listener to be significantly more adequate to the proposed speaking styles than the traditional neutral voices.

2. Speech Corpora and Average Models

For both the speaking styles and neutral read speech corpora we used a combination of pre-existent databases and some new recorded hand-labeled data, separated as follows:

2.1. Speaking Styles Speech Data

C-ORAL-Rom Database A multi-language multi-style database [6]. Out of all the available data, only three of the styles available in the Spanish formal media section were used: news, sports and interviews, and between all the available data of each style a subset of the least noisy audio files was selected.

TC-STAR run 3 A multi-language database of recorded parliamentary speeches in different environments [7] such as the European Parliament or the Spanish Parliament. Out of all the available data, four different speakers in the Spanish Parliament subsection were used.

Self Labeled Data Because some of the styles did not amount to enough data (namely: news and sports), additional speech was processed and added to the models. For the news style, recorded data of live news by a very famous Spanish newscaster was processed. Finally, for the sports commentary style, we aligned and labeled 15 minutes of the broadcast of the Eurocup2012 finals.

2.2. Neutral Read Speech Data

UVIGO-ESDA Database A database consisting of a single male amateur Spanish speaker (UVD) in a neutral read speech situation for approximately 2 hours of speech recorded in studio [8]. This speaker was used for obtaining both the average modeling and also for the implementations of the speaking styles extrapolation techniques.

SEV Database An emotional database consisting of a male and a female speaker [9]. Only the neutral speech of the male speaker was used, and only for the average modeling.

New Recorded Data In order to increase the variability of the neutral read speech data we also added a few speakers of those previously recorded in our laboratory environment. The recording is done inside an acoustically-treated room, so the obtained quality is very high. Out of all this data, 4 male speakers were added to the average model data pool and 2 of them, one possessing a mid-range pitch (JLC) and a final one with a high-range pitch and a soft Colombian accent (JEC) were used for the final synthesis and perceptual test.

With all the mentioned data and by using Speaker Adaptive Training (SAT) [4] a complete background average model was obtained. This model, which contains both neutral and speaking styles data, will be used as the basis of all further adaptations, significantly reducing the final voices training time and increasing the overall quality and robustness of the models as proved in some of our previous work [10].

2.3. Adapted Average Models

As the full background model is too complex to be able to capture particular nuances of the different expressive styles, we applied an intermediate adaptation step with which we obtained average models of the 4 speaking styles (sports, news, interviews, politics) and an additional one for the neutral read speech speakers. This adaptation was done by using the CSMAPLR algorithm [3], chosen because of its synergy with SAT models and high quality adaptation even with the small adaptation data available for the speaking styles models. The biggest advantage of having average models of the 5 speaking styles considered is that the differences between the pure styles can be characterized and exploited for the expressiveness transplantation, as will be exploited and explained later in this paper.

Finally, the speaker models of the neutral speakers used for the perceptual test were adapted using CSMAPLR again but in this case directly adapting from the neutral average, which will be an important detail when facing the speaking styles adaptation through transplantation that will be defined in the next section.

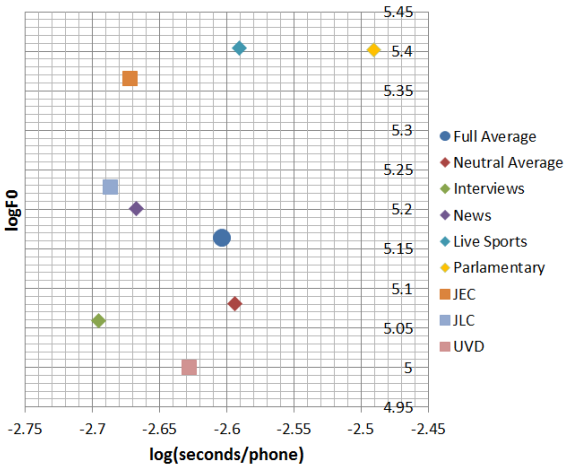


Figure 1: Prosody characterization of the training dataset. Diamonds represent the average models and squares the neutral speakers. Uttering speed was selected to obtain the faster models to the right while keeping them separable.

2.4. Analysis of the Training Data

In figure 1 we can see a plot of two main prosody features (logF0 and -1/uttering speed) of the data used to train the average models. In it we can define 3 F0 bands around the 3 test speakers, a low-range F0 for UVD, mid-range F0 for JLC and a high band for JEC. Similarly we can consequently associate also each style to a band: interviews to low-F0, news to mid-range F0 and live sports and parliamentary speech to high F0. This was already observed in a previous study [11]. That is, a live sports commentary or a parliamentary speech will be typically recorded in an open, noisy environment while an interview or a newscast will be recorded on studio. Uttering speed can be explained by the spontaneity of the style: a newscaster will have a prepared script that can be read quickly while a politic will tend to somewhat improvise on the reactions of the public.

3. Speaking Styles Transplantation

The main objective of the present research is to be able to transplant the nuances of particular speaking styles speech models into a different neutral speaker. This would imply a significant step up in the availability of task-dependent voices, which is much needed when considering naturalness of the synthetic speech. Additionally, because the proposed techniques are based on adaptation the harvesting of data becomes much simpler, as it is possible just to pick speaking styles examples of a particular style or emotion of different speakers from any source in order to train the average models used as the basis for the extrapolation. This in the end means a substantial increase in the ease of producing synthetic voices with speaking styles.

3.1. Proposed Approach: Transplantation through Adaptation

One of the biggest advantages of parametric speech synthesis is its versatility, and adaptation is a technique that exploits that versatility successfully. It can then be used to obtain robust models from a background average and a few minutes of the speaker, and also to obtain the transformation functions between models to help characterize the differences between them.

This principle was applied in our Albayzin2012 speech synthesis challenge submission [10] to successfully control the expressive strength of emotional models by assuming that the transformation relating the expressive and neutral model can be scaled, allowing for a linear continuous modeling of the expressiveness space.

Following up on the strength control concept the concept of expressiveness extrapolation appears. If the transformation function between an expressive model and a reference model can be transferred to a different speaker, it is natural to think that the expressiveness will be likewise transferred to the target speaker.

It is not acceptable to think that the relationship between a particular speaker's expressive representation and that same speaker's neutral voice is the real representation of that expressiveness, and that is why we propose the use of averages. If we can model the transformation between the target speaker and the neutral average and apply this transformation to the expressive average, it is to be expected that we will obtain the desired target expressive speaker (figure 2).

Both the adaptation process between the background averages and the adaptation between the neutral average and the neutral target speaker are done using CSMAPLR. This presents the advantage of providing linear transformations that consider both mean and variances. If the adaptation were not to be constrained the variances would be ignored and the expressive nuances could be lost [3], and the linearity of the transformation reduces the complexity of the modeling.

Both CSMAPLR adaptation transformations can be expressed as :

$$\bar{\mu}_{exp} = \zeta_{exp}\mu_N + \epsilon_{exp} \quad (1)$$

$$\bar{\Sigma}_{exp} = \zeta_{exp}\Sigma_N\zeta_{exp}^T \quad (2)$$

$$\bar{\mu}_{spk} = \zeta_{spk}\mu_N + \epsilon_{spk} \quad (3)$$

$$\bar{\Sigma}_{spk} = \zeta_{spk}\Sigma_N\zeta_{spk}^T \quad (4)$$

Where $\bar{\mu}_{exp/spk}$ and $\bar{\Sigma}_{exp/spk}$ are the target means and covariance matrices of the expressive and target speaker models respectively, with ζ defining the rotation matrix and ϵ the bias that are obtained following the CSMAPLR algorithm [3]. Consequently, the transplantation transform is defined as follows:

$$\bar{\mu}_{tra} = \zeta_{spk}\zeta_{exp}\mu_N + \zeta_{spk}\epsilon_{exp} + \epsilon_{spk} \quad (5)$$

$$\bar{\Sigma}_{tra} = \zeta_{spk}\zeta_{exp}\Sigma_N\zeta_{exp}^T\zeta_{spk}^T \quad (6)$$

3.2. Alternatives to Transplantation: Copying the Speaking Style Average Model

In order to test the relevance of the proposed transplantation adaptation technique we defined a set of alternatives that could be considered for expressive synthesis, namely copying the different feature streams from the average models into the target speaker model. In this case it meant copying either the prosody features (F0 and duration streams) or the spectral features. This would not be an easy thing to do in a situation in which every voice was trained independently following the traditional HMM-based modeling, as the decision trees would not be shared. But because our voices share a common background model and the adaptation process keeps the trees intact, copying the prosody or the spectrum from the style averages is as simple as replacing the desired model files in the target speaker.

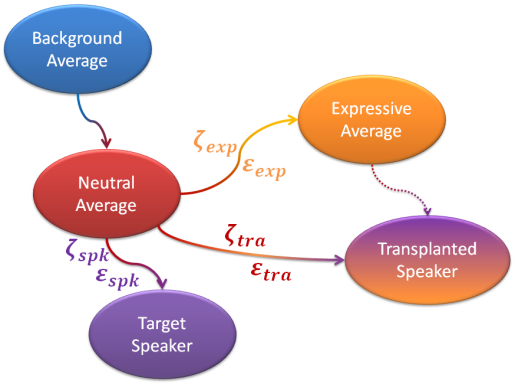


Figure 2: Schematic of the extrapolation through adaptation system.

3.2.1. *Copying the Prosody*

Prosody is known to carry a very significant portion of the expressive load of speech [12]. As such it is assumable that by simply copying the prosody (only F0 and phone durations are considered) of a clear representative of a speaking style model would yield acceptable extrapolation results at least if the target speakers' F0 and the average's F0 are not too dissimilar. On the other hand, it is also possible that if this last condition does not hold the quality of the output voice would degrade or that instabilities might appear.

3.2.2. *Copying the Spectrum*

Spectrum is assumed to hold most of features related to the identifiability of the speaker, although it also includes some expressive features [12]. In that sense it is safe to assume that this approach would not be very successful for extrapolating the expressiveness of the model but instead copy the identity of the style's average. This by itself does not seem useful for the task at hand, although for example in some extreme cases were a speaker can be clearly associated to a particular style it would fulfill a similar purpose. Nonetheless, it seems interesting to test the results this kind of approach would give.

4. Perceptual Test Description

To test the effectiveness of the proposed technique we prepared a web-based perceptual test in which 32 listeners were asked to establish two different rankings in order of preference: adequacy of the speaking style to the synthesized text and similarity to the original speaker. The decision to make the test ranking-based was taken because, as the task is considerably difficult and there is no natural voice reference available for the listener, comparing the system between themselves instead of assigning a value to them facilitated the testing process.

The target's speaker neutral voice and the speaking style average model were synthesized to be added as top-line systems: the neutral voice would provide the top-line for the similarity analysis and the average model for the adequacy of the speaking style task.

The test consists of 5 systems (top-line systems, transplantation system, copy-prosody system and copy-spectrum system) and 4 styles (news, sports, interviews and parliamentary speech) for each of the 3 evaluated target speakers.

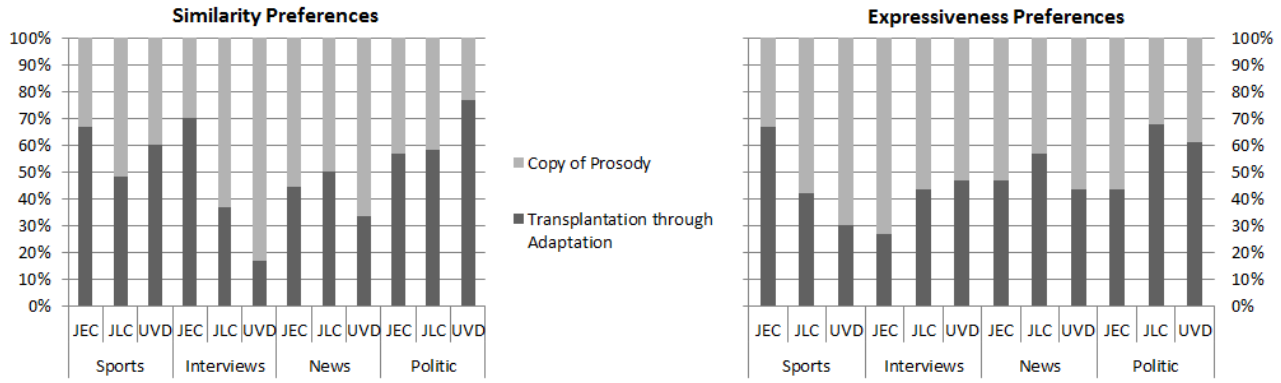


Figure 3: Results of the perceptual test comparing the user preferences between transplantation and copy of prosody for all test speakers and styles.

Regarding the interface, the listeners were presented with all five stimuli at the same time so that they could be played as many times as necessary. The stimuli distribution was designed following a balanced latin square randomization of the questions pattern, resulting in a total of 8 utterances required per style. The utterances were extracted from real media texts and were selected so as to be longer than 10 words for the phrase-level prosody to become relevant. Finally, the minimum acceptable number of tests per target speaker was decided to be 16: two complete rounds of tests.

5. Results

An initial consideration verified by table 3 is that copying the spectrum of the average or using the average itself is not a valid transplantation technique because these methods do not keep the identity of the target speaker, even if the perceived adequacy is comparable between the 5 proposed systems (table 2). As such the analysis will focus only on when is adaptation considered by the listeners better than copy of prosody and vice versa.

The first result that becomes evident from both table 1 and figure 3 is that there is no significant difference overall between transplantation and extrapolating by copying the prosody of the average model. Even so, in a global level it can be seen that specially the parliamentary speech style but also the live sports commentary (the higher F0 ones, as seen in figure 1) favor the transplantation-based system while news favors the copy of prosody in the similarity department, without relevant differences in adequacy. This is reinforced by the similarity preference results of JEC, the high pitched target speaker, for which the results definitely show that transplantation through adaptation is the preferred technique.

The consideration to be drawn from the adequacy results is that even if in average the test results do not favor any of the techniques, the more different speaker-style pair of models (i.e. UVD with politics or JLC with politics) also appear to be more adequate for transplantation than for copy of prosody. This could be seen as a hint that while copying the prosody is an acceptable method of extrapolating the speaking style, it stops being reliable when the pair of models is too disparate in prosody. On the other hand, transplantation through adaptation does not fall off in these kind of situations because not only the prosody is adapted but also the spectrum of the models, preventing instabilities or unnatural sounds from appearing.

Table 1: Number of utterances preferred (>) by listeners in terms of expressiveness and similarity between transplantation-based (Trans) and copy prosody-based (C-Pro) systems.

Expressiveness	SIMILARITY		TOTAL
	Transp<C-Pro	Transp>C-Pro	
Transp<C-Pro	**98**	**91**	**189**
Sports	22	27	49
Interviews	30	25	55
News	29	17	46
Politic	17	22	39
Transp>C-Pro	**76**	**98**	**174**
Sports	16	26	42
Interviews	23	12	35
News	21	23	44
Politic	16	37	53
TOTAL	**174**	**189**	**363**

6. Conclusions

The first conclusion that can be drawn from the test is that the extrapolation of speaking styles can provide synthetic voices more adequate to different tasks (i.e. style of delivery) than simpler neutral voices for any speaker without requiring the target speaker to record any non-neutral data. This is a huge step-up from the traditional synthesis algorithms that would require the target speaker to record a new database for every expressive realm they want their voice on.

Also, we have seen that using average models allows this extrapolation to be done with as little as 15 minutes of speaking styles data. It has been done both by copying these average models' prosody or by applying the more advanced technique of adapting between neutral and speech with speaking styles to the neutral speaker. In general both techniques appear to be capable of imbuing the target voices with speaking styles while keeping the source identity, but we have found a trend in which target voices that are too different from the average models start producing worse quality voices when just copying prosody.

Additionally, when considering different applications such as emotional speech synthesis, merely copying the prosody will not be able to extrapolate the expressiveness in all situations, requiring a more complex approach such as our transplantation process.

Table 2: Results in adequacy ranking for the different systems averaged between the 3 target speakers.

ADEQUACY	Read Speech	Style Average	Copy of Spectrum	Copy of Prosody	Transplantation
Sports	2.57	2.50	2.15	3.96	3.81
Interviews	2.63	3.69	2.98	2.94	2.76
News	2.98	3.22	2.68	3.37	2.76
Politic	3.49	2.82	2.40	3.02	3.26
Average	2.92	3.06	2.55	3.32	3.15

Table 3: Results in similarity ranking for the different systems averaged between the 3 target speakers.

SIMILARITY	Read Speech	Style Average	Copy of Spectrum	Copy of Prosody	Transplantation
Sports	4.31	1.85	2.36	3.13	3.35
Interviews	4.50	1.68	2.28	3.37	3.18
News	4.36	1.89	2.12	3.41	3.22
Politic	4.40	1.73	2.35	3.01	3.51
Average	4.39	1.79	2.28	3.23	3.32

Even so, the results are not significant enough yet, so the planned future work is two-fold: first of all increase the available training data for each speaking style so as to obtain much more informative averages from which to adapt. We also plan to add strength control capabilities to the transplantation adaption system in order to try different control ratios to try and find optimal control values that increase the perceived adequacy of the style by enhancing particular features that carry more expressiveness information. Finally, we intend to test the proposed system in an emotional environment and compare it with the considered systems once again to verify the versatility we can provide.

7. Acknowledgments

The work leading to these results has received funding from the European Union under grant agreement 287678. It has also been supported by TIMPANO(TIN2011-28169-C05-03), INAPRA (MICINN, DPI2010-21247-C02-02), and MA2VICMR (Comunidad Autonoma de Madrid, S2009/TIC-1542) projects. Jaime Lorenzo has been funded by Universidad Politecnica de Madrid under grant SBUPM-QTKTZHB. Authors also thank the other members of the Speech Technology Group and Simple4All project for the continuous and fruitful discussion on these topics.

8. References

[1] Rob Clark and Simon King, "Simple4all - http://simple4all.org," 2011.

[2] T. Yoshimura, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, "Simultaneous modeling of spectrum, pitch and duration in hmm-based speech synthesis," in *Sixth European Conference on Speech Communication and Technology*, 1999.

[3] J. Yamagishi, T. Kobayashi, Y. Nakano, K. Ogata, and J. Isogai, "Analysis of speaker adaptation algorithms for hmm-based speech synthesis and a constrained smaplr adaptation algorithm," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 17, no. 1, pp. 66–83, 2009.

[4] J. Yamagishi and T. Kobayashi, "Average-voice-based speech synthesis using hsmm-based speaker adaptation and adaptive training," *IEICE TRANSACTIONS on Information and Systems*, vol. 90, no. 2, pp. 533–543, 2007.

[5] Langzhou Chen, Mark Gales, Vincent Wan, Javier Latorre, and Masami Akamine, "Exploring rich expressive information from audiobook data using cluster adaptive training," in *Interspeech 2012, 13th Annual Conference of the International Speech Communication Association. Portland, Oregon. September 9-13*, 2012.

[6] A. Moreno-Sandoval, G. De la Madrid, M. Alcántara, A. Gonzalez, JM Guirao, and R. De la Torre, "The spanish corpus," *C-ORAL-ROM: Integrated Reference Corpora for Spoken Romance Languages, Amsterdam: John Benjamins Publishing Company*, pp. 135–161, 2005.

[7] A. Cardenal-Lopez L. Docio-Fernandez and C. Garcia-Mateo, "Tc-star 2006 automatic speech recognition evaluation: The uvigo system," pp. 145–150, 2006.

[8] C.G. Mateo E.T. Banga, "Documentation of the uvigo-esda spanish database," Tech. Rep., Grupo de Tecnoloxias Multimedia, Universidad de Vigo, Vigo, Espaa, 2010.

[9] R. Barra-Chicote, J. M. Montero, J. Macias-Guarasa, S. Lufti, J. M. Lucas, F. Fernandez, L. F. D'haro, R. San-Segundo, J. Ferreiros, R. Cordoba, and J. M. Pardo, "Spanish expressive voices: Corpus for emotion research in spanish," *Proc. of LREC*, 2008.

[10] Jaime Lorenzo-Trueba, Oliver Watts, Roberto Barra-Chicote, Junichi Yamagishi, Simon King, and Juan M. Montero, "Simple4all proposals for the albayzin evaluations in speech synthesis," in *Iberspeech2012, VII Jornadas en Tecnologia del Habla and III Iberian SLTech Workshop*, 2012.

[11] Jaime Lorenzo-Trueba, Roberto Barra-Chicote, Tuomo Raitio, Nicolas Obin, Paavo Alku, Junichi Yamagishi, and Juan M Montero, "Towards glottal source controllability in expressive speech synthesis," in *Interspeech 2012, 13th Annual Conference of the International Speech Communication Association. Portland, Oregon. September 9-13*, 2012.

[12] Roberto Barra-Chicote, *Contributions to the analysis, design and evaluation of strategies for corpus-based emotional speech synthesis*, Ph.D. thesis, ETSIT-UPM, 2011.

Investigating the shortcomings of HMM synthesis

Thomas Merritt, Simon King

The Centre for Speech Technology Research, The University of Edinburgh, U.K.

T.Merritt@ed.ac.uk, Simon.King@ed.ac.uk

Abstract

This paper presents the beginnings of a framework for formal testing of the causes of the current limited quality of HMM (Hidden Markov Model) speech synthesis. This framework separates each of the effects of modelling to observe their independent effects on vocoded speech parameters in order to address the issues that are restricting the progression to highly intelligible and natural-sounding speech synthesis.

The simulated HMM synthesis conditions are performed on spectral speech parameters and tested via a pairwise listening test, asking listeners to perform a "same or different" judgement on the quality of the synthesised speech produced between these conditions. These responses are then processed using multidimensional scaling to identify the qualities in modelled speech that listeners are attending to and thus forms the basis of why they are distinguishable from natural speech.

The future improvements to be made to the framework will finally be discussed which include the extension to more of the parameters modelled during speech synthesis.

Index Terms: Speech synthesis, Hidden Markov models, Vocoding

1. Introduction

Despite several years of improvements in the quality of speech generated using HMM (Hidden Markov Model) synthesis, this type of synthetic speech still stubbornly remains significantly less natural than speech output from good concatenative (unit selection) synthesis systems [1, 2], as consistently reflected in the results from the annual Blizzard challenge [3, 4, 5]. Although it can achieve higher intelligibility than unit selection, HMM synthesis is not yet as natural as unit selection, and neither are judged by listeners to be as natural as real speech.

It is common in the literature to find the cause for the reduced naturalness of HMM speech stated as "over-smoothing", and that this is the fault of the statistical model, but to the best of our knowledge there are no formal, published studies supporting this claim. The idea of "over-smoothing" is at first glance seemingly a simple one, but may conflate a number of different effects of signal representation and of statistical modelling in both spectral and temporal domains. Smoothing is inherent in the statistical modelling framework, of course. The spectral envelope is smoothed first by the low-dimensional representation, then again by averaging over consecutive frames and over multiple tokens. The temporal structure of the speech parameters is smoothed because the model represents the trajectory with limited resolution (e.g., 5 states per phone-sized-unit).

What is needed is a framework in which we can separate out the different contributions of the various processes of modelling. This is the contribution of this paper.

1.1. A simulation framework

This paper introduces such a framework and – as a first illustration of its use – tests a couple of the potential causes of the degradation in naturalness introduced by the use of statistical models. The framework is general and could be applied to many different aspects of the problem. The idea is to *simulate* the effects of modelling vocoded speech, in a carefully controlled manner. Knowledge obtained by such experiments could then be used to identify those areas that are causing the problem, and to eventually rectify them.

Current HMM-based synthesisers are large, complex systems. There are interactions between the signal processing (e.g., how the spectral envelope is extracted and how it is represented for the purposes of modelling) and the modelling (e.g., the parameter sharing structure of the model and how much data are available to estimate each free parameter) which need to be investigated. In the work presented here, this will be done by removing the modelling part completely and replacing it with a series of operations which are designed to simulate some modelling effects. Our proposed approach allows us to vary the strength of these effects, and to examine the interactions between them. Thus, by using simulation, we can continuously vary the system from being a simple vocoder at one end of the scale, to a simulated HMM synthesiser at the other. In this paper, the effects that we use are temporal smoothing and variance scaling of the speech parameters representing the spectral envelope.

1.2. Measuring the effects

The second component of the proposed framework is perceptual testing of the acoustic consequences of the simulated effects of statistical modelling. Asking listeners to attend to specific aspects of the speech is problematic [6, 7] and also risks biasing them towards certain phenomena. Since we are not entirely sure what perceptual dimensions listeners use when rating the naturalness of synthetic speech, it is not clear what aspects of the signal we could ask them to attend to. Therefore, we adopt a less direct methodology, and ask the listeners to perform a very simple task where the instructions contain no bias towards any particular acoustic property or perceptual dimension. This task is a simple "same or different" judgement on pairs of stimuli, from which we can derive a matrix of pairwise perceptual distances. Multidimensional scaling (MDS) allows such data to be visualised and from this visualisation we can identify the perceptual dimensions, that is, what the listeners are attending to. Tracing these back to the simulated effects involves interpreting the MDS visualisation.

1.3. Structure of this paper

Section 2 will discuss how we implemented a simulation of HMM synthesis, section 3 will introduce the method for perceptually testing the speech created under this simulation, then section 4 presents the results from this testing. Based on these results, we offer an interpretation and some conclusions in section 5 followed by a summary of the contributions of this paper. Finally, section 6 will suggest future work, including how we plan to use the proposed framework to simulate many more of the effects of statistical modelling.

2. Methodology

Our aim is to tease apart the complex effects of statistical modelling on synthetic speech. In order for the contributing factors (to shortcomings in the quality of speech output by HMM synthesis) to be investigated, we need a framework in which these effects can be individually manipulated – a kind of 'oracle' HMM synthesiser which allows for complete control over each aspect of the system, varying it between some form of 'ideal', or 'perfect' component and the real component used in a full HMM synthesiser. An obvious example of the 'ideal' is a vocoder, which has access to natural speech parameters and is so unaffected by any flaws in the way the statistical modelling part reconstructs these.

2.1. Scope of the current investigation

In the present work, we concentrate on global simulations of the statistical modelling part of the system. This is illustrated in figure 1, where we can see that the speech parameter extraction and waveform generation (reconstruction) parts are the same as in a full HMM synthesiser. Extraction of the spectral, F0, and aperiodic energy speech parameters is performed as usual, with the use of STRAIGHT (Matlab implementation)[1] [8, 9] followed by SPTK [10] to convert the spectral envelope to line spectral frequencies (LSFs), F0 to log F0 and aperiodic energy to band aperiodic energy. We chose to use LSFs because they are more convenient for visualisation than, say, Mel-generalised cepstra, and this should ease the interpretation of the results later. The conversion of F0 to log F0 and aperiodic to band aperiodic was also performed to simulate common modelling conditions of all speech parameters, this allows us to better track the effect that modelling has on the spectral envelope parameters by implementing a system which is more realistic. We also focus only on the spectral envelope speech parameters here; experimentation with the other speech parameters is future work.

Following the application of our modelling simulations, the LSFs, log F0 and band aperiodic energy parameters were converted back into spectral, F0 and aperiodic energy speech parameters using SPTK [10] before performing the 'reconstruction' phase of HMM speech synthesis, by inputting the speech parameters into STRAIGHT (Matlab implementation) to obtain the synthesised speech waveform as output.

2.2. Simulating "over-smoothing"

There are several ways in which the output speech parameters of an HMM synthesiser are "too smooth". Here, we concentrate on temporal effects, leaving spectral smoothness as future work. Looking at the output of typical HMM systems [2, 11], we generally find far less temporal detail than is observed in the speech parameters for natural speech. Some of this detail may simply be noise introduced by the spectral envelope estimation process, but some of it may be perceptually important. We investigate this by temporally smoothing the speech parameters, which simulates the limited temporal resolution of 5-state-per-phone models and the subsequent MLPG [1, 12] trajectory generation algorithm.

Another consequence of statistical modelling is that the variance of the generated speech parameters is lower than those from natural speech. This has long been known to significantly reduce the quality of the generated speech and is why mitigating this by considering Global Variance (GV) [13, 2] has such as dramatic positive effect on quality. However, GV cannot guarantee to perfectly restore the correct variance of the parameters. We simulate the effect of modelling and of GV by scaling the standard deviation of the speech parameters by a value greater or less than 1.0.

Removing temporal detail via smoothing will also slightly reduce the variance of the speech parameters. We can examine the interaction between temporal smoothness and variance by applying both effects, with varying strengths. It is worth repeating at this point that temporal smoothing and variance scaling are certainly not a comprehensive simulation of HMMs synthesis, but that they were used here as a starting point for an ongoing investigation and that more complex effects will be investigated in future work.

The effects simulated in the current work are all applied to each speech parameter independently and are implemented utterance-by-utterance.

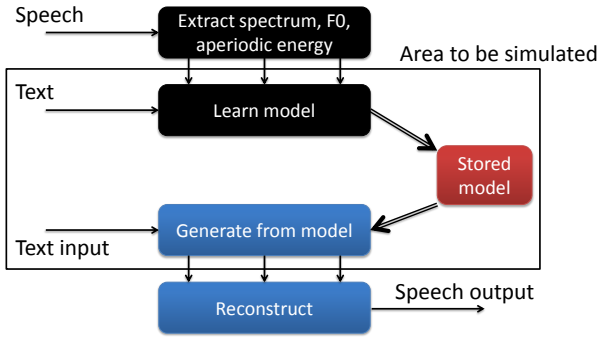


Figure 1: *Training and using an HMM speech synthesiser, illustrating the part of the process that is simulated here.*

2.2.1. *Temporal smoothing*

The smoothing effect was implemented as a weighted moving average, sliding a Hanning window over the signal (i.e., each LSF in turn), to simulate the limited temporal resolution of HMM modelling. The width of the window was varied, to impose varying amounts of smoothing.

2.2.2. *Variance scaling*

Variance adjustment was implemented as a simple scaling of the standard deviation by a fixed factor. For each parameter (i.e., each LSF) in turn, the mean value over the utterance

[1] STRAIGHT V40_007 methods were used, these were written by Hideki Kawahara

was found and subtracted before multiplying the parameter by a scalar value, and finally adding the mean back in. By altering the scalar value, the standard deviation is correspondingly adjusted, to simulate both reduced variance (which is commonly observed in HMM synthesis) and increased variance (e.g., as may happen if a Gaussian p.d.f. is poorly estimated during training, or when GV fails to re-instate the appropriate amount of variance).

3. Experiments

A range of simulated effects were selected to be tested, with the strengths of modifications being selected by informal listening to reflect the sorts of imperfections we have ourselves encountered in many of the HMM synthesis systems we have built. For the temporal smoothing, Hanning window sizes of 80 and 110 frames (at a frame rate of 5 msec) were selected, along with a 'no smoothing' condition. Smaller window widths (i.e., less smoothing) were found to produce negligible perceptual effects. Variance adjustment involved scaling the standard deviation by scalar values of 0.6, 0.8, 1.2 and 1.4 as well as a 'no variance adjustment' condition equivalent to scaling by 1.0. These particular values for smoothing and variance adjustment were selected to provide audibly different speech quality, whilst staying within the range of qualities that we have observed in real HMM synthesisers.

3.1. Materials

The speech corpus used for testing was a set of Harvard Sentences [14] read by a male professional speaker of British English (known as 'Nick' and whose speech has been used in the Hurricane Challenge [15] and who also features in the 'mngu0' acoustic-articulatory corpus[2] [16]), this was sampled at 16 KHz. The methodology for preparing the stimuli was, as described above, to extract speech parameters using STRAIGHT and SPTK, to apply the two simulated effects of smoothing and variance adjustment with all possible combinations of strengths including the 'no modification' conditions, then to reconstruct the waveform. Order 30 LSF coefficients were used as this offers a good representation of the spectral information for the speech at the sampling rate used. The result was $3 \times 5 = 15$ versions of each of 40 sentences.

The variance adjustment method was applied per speech parameter per utterance independently, so the mean speech parameter value subtracted before scaling is influenced by the amount of silence present; therefore, the material was manually edited to leave only just a few 100 msec of leading and trailing silence. Care was also taken to remove any background noise present during the non-speech, because in preliminary experiments this became perceptually much more apparent after applying some of modifications.

3.2. Listening test

In the listening test, listeners had to make forced choice 'same or different quality' judgements about pairs of stimuli.

The testing was performed by applying each of the 15 simulation conditions (called A to O) as defined in table 2, which combine smoothing and/or variance adjustment to each of the 40 sentences. The 40 sentences were divided into 20 pairs (sentences 1 & 2, sentences 3 & 4, and so on), and for each of these pairs of sentences, all possible combinations of conditions (e.g., sentence 1 in condition A + sentence 2 in condition F) were created, except for pairs of identical conditions (e.g., sentence 1 in condition A + sentence 2 in condition A), as shown in figure 3.

This resulted in $20 \times ((15 \times 15) - 15) = 4200$ pairs of sentences, which were then randomised in order and divided amongst 30 listeners, resulting in each listener listening to 140 pairs of sentences and thus making 140 'same or different' judgements. These listeners were selected at random from applicants to an online advert placed in the University of Edinburgh's Student And Graduate Employment service; all were native English speakers with no self-reported hearing problems. The stimuli pairs were presented in a randomised order per listener over high quality headphones in quiet sound-proofed booths with no distractions.

[2] http://www.mngu0.org

Condition index	Hanning smoothing window size	Standard deviation scaling
A	none	0.6
B	80	0.6
C	110	0.6
D	none	0.8
E	80	0.8
F	110	0.8
G	none	none
H	80	none
I	110	none
J	none	1.2
K	80	1.2
L	110	1.2
M	none	1.4
N	80	1.4
O	110	1.4

Figure 2: *The 15 conditions combining each level of smoothing (including no smoothing) and each amount of standard deviation scaling (including no modification)*

		Sentence 1														
		A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
Sentence 2	A	×	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	B	✓	×	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	C	✓	✓	×	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	D	✓	✓	✓	×	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	E	✓	✓	✓	✓	×	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	F	✓	✓	✓	✓	✓	×	✓	✓	✓	✓	✓	✓	✓	✓	✓
	G	✓	✓	✓	✓	✓	✓	×	✓	✓	✓	✓	✓	✓	✓	✓
	H	✓	✓	✓	✓	✓	✓	✓	×	✓	✓	✓	✓	✓	✓	✓
	I	✓	✓	✓	✓	✓	✓	✓	✓	×	✓	✓	✓	✓	✓	✓
	J	✓	✓	✓	✓	✓	✓	✓	✓	✓	×	✓	✓	✓	✓	✓
	K	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	×	✓	✓	✓	✓
	L	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	×	✓	✓	✓
	M	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	×	✓	✓
	N	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	×	✓
	O	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	×

Figure 3: *One set of pairings of sentences and conditions in the listening test.*

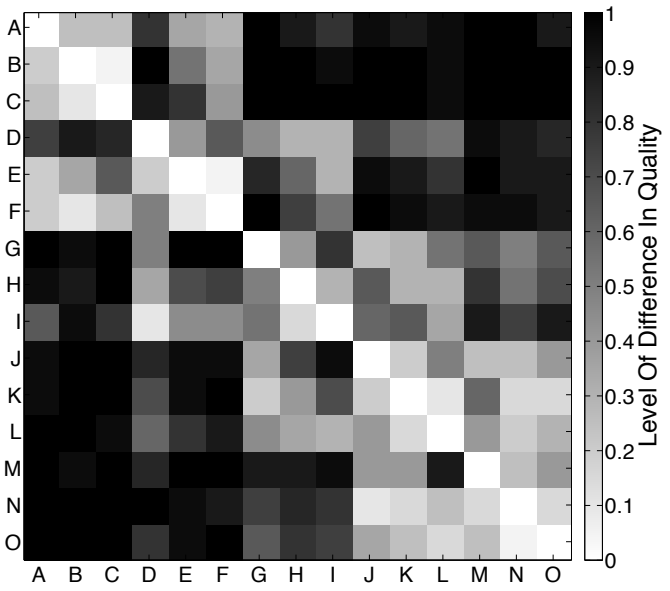


Figure 4: *Listeners' responses between conditions presented in figure 2, pooled across all sentences and listeners. Darker shades indicate greater perceived dissimilarity between conditions.*

3.3. Multidimensional scaling

The raw listener responses were pooled across all listeners and all sentences for each individual combination of modifications. The result is a dissimilarity matrix, in which each cell contains a number indicating the perceived dissimilarity between two conditions. Figure 4 shows this matrix graphically: each cell contains the number of comparisons between a pair of conditions marked as 'different' by listeners. Multidimensional scaling was used to analyse this matrix, and create a plot in which each condition appears as a point. Short distances between points on the plot indicate perceptual similarity and large distances indicate dissimilarity [17]. We used a Matlab implementation of MDS based on Kruskal's normalised STRESS1 criterion[3].

4. Results

MDS projects the dissimilarity matrix into a multi-dimensional space. In order to find an appropriate dimensionality of this space, one must compromise between accuracy of representation (in higher dimensions, the correspondence between dissimilarity and distance in the space will be more precise) against the need for a modest number of dimensions to allow for the data to be visualised and for the axes to be interpreted. The so-called stress value computed as part of the multidimensional scaling algorithm reflects this tradeoff; figure 5 plots the stress value for various dimensionalities. It seems that three dimensions is a reasonable operating point for our data.

The first two dimensions of the three-dimensional space found by multidimensional scaling is given in figure 6. Distance in this space indicates perceived dissimilarity: the closer a point is to the natural unmodified speech, the "more natural" it sounds. It is immediately apparent that the listeners judgements cannot be explained by a single dimension and that they

[3]function 'mdscale' from the Matlab statistics toolbox

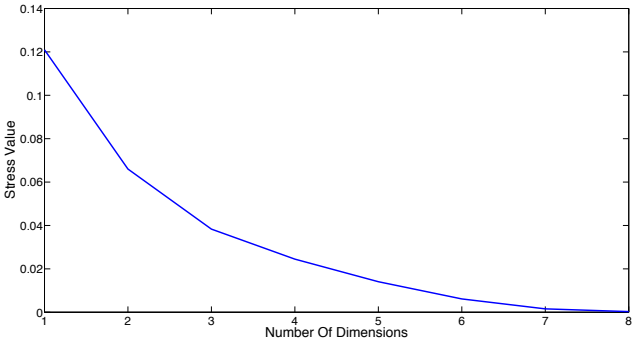


Figure 5: *Stress levels returned by MDS at different dimensions.*

are making their decisions based on more than one aspect of the speech:

- The horizontal axis seems to relate to the amount of LSF variance, with the reduced variance speech clearly different from the increased variance speech
- The vertical axis seems to relate to overall quality of synthesis, regardless of the LSF variance, with both reduced and increase variance speech being placed towards the top of the space, whereas natural speech is at the bottom.

This plot also shows that the smoothing has only a secondary effect, probably simply because it has the side effect of slightly reducing variance. When the variance is too high (right hand side of figure 6), then the smoothing has a beneficial effect, moving the points lower and therefore closer to natural speech.

5. Conclusions

We have introduced a simple-to-use, extensible methodology that can tease apart the contributions to speech quality of the various components of an HMM-based text-to-speech system. The fundamental idea is to simulate all or part of the system, and thus to gain explicit control over the system's behaviour. In this paper, we have demonstrated the use of this framework in a straightforward way, by simulating a complete HMM-based synthesiser as simply a combination of smoothed parameter trajectories and incorrect variance.

Even from this very simple simulation, we can conclude that listeners are able to perceive different types of quality reduction: the MDS analysis reveals that they can make overall quality judgements (vertical axis of figure 6) and at the same time clearly distinguish whether this is due to too high or too low variance. It also seems fairly safe to conclude that *temporal* smoothness in LSF trajectories is not really a problem and leads to only very small perceptual effects.

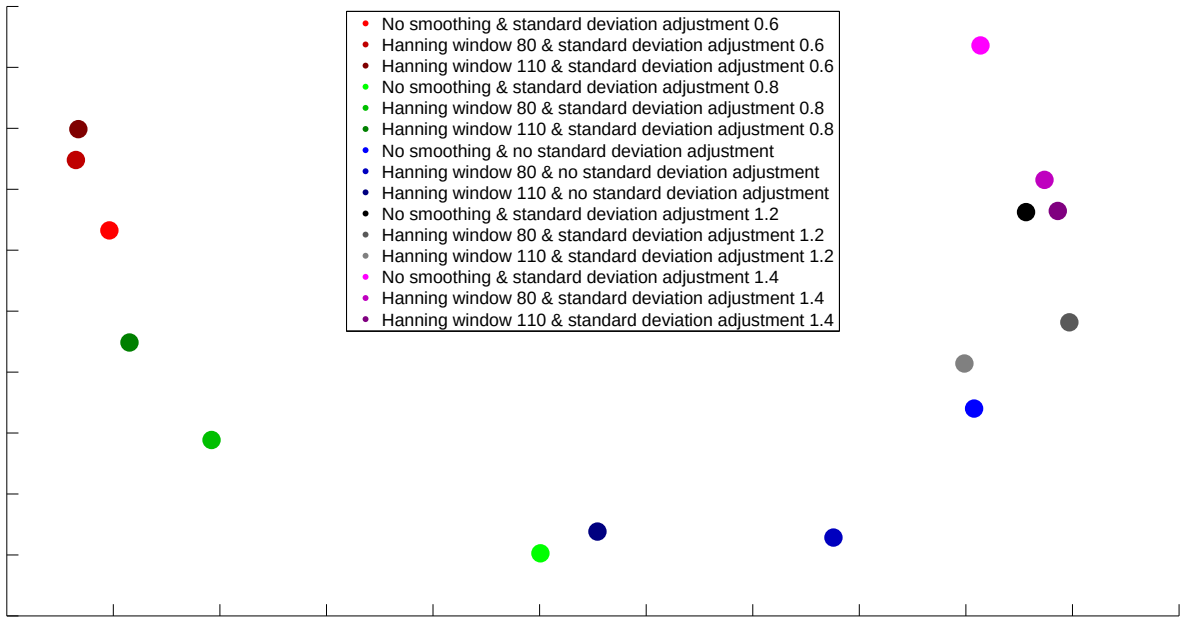


Figure 6: *Plot of the first two dimensions of the three-dimensional space found using MDS.*

6. Future work

The next steps are obvious: to extend the range of simulated effects of modelling and conduct further listening tests followed by MDS analysis of the responses. The ultimate aim is a system that can be continuously controlled between an 'oracle' vocoder and a fully-modelled text-to-speech system. Some categories of effects that we would like to simulate next include:

- *spectral envelope* over-smoothness: formant dulling and sharpening; suppression or emphasis of spectral detail
- averaging across *multiple tokens* of similar speech sounds (e.g., phonemes in context) at frame, state and model granularities
- poor modelling of the *covariance* within a set of speech parameters (e.g., LSFs), resulting in inconsistent sets of values
- *inconsistencies* between the different speech parameter streams (e.g., aperiodic energy vs. spectral envelope) caused by use of different model parameter tying structures
- model boundary *discontinuities* in the trajectory (which may be disguised but not overcome by MLPG) occurring at transitions between HMMs of phoneme-sized units

7. Acknowledgements

This work was supported by EPSRC Programme Grant EP/I031022/1 (Natural Speech Technology). Thanks also to Heng Lu for his assistance with STRAIGHT and SPTK, Rob Clark for his advice on MDS and Cassie Mayo for her advice on perceptual testing and experimental design.

8. References

[1] S. King, "An introduction to statistical parametric speech synthesis," *Sadhana*, vol. 36, no. 5, pp. 837–852, 2011.

[2] H. Zen, K. Tokuda, and A. W. Black, "Statistical parametric speech synthesis," *Speech Communication*, vol. 51, no. 11, pp. 1039 – 1064, 2009. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167639309000648

[3] S. King and V. Karaiskos, "The Blizzard Challenge 2011," in *Proc. Blizzard Challenge*, Turin, Italy, 2011.

[4] ——, "The Blizzard Challenge 2010," in *Proc. Blizzard Challenge*, Kansai Science City, Japan, 2010.

[5] ——, "The Blizzard Challenge 2009," in *Proc. Blizzard Challenge*, Edinburgh, United Kingdom, 2009.

[6] C. Mayo, R. A. Clark, and S. King, "Listeners weighting of acoustic cues to synthetic speech naturalness: A multidimensional scaling analysis," *Speech Communication*, vol. 53, no. 3, pp. 311–326, 2011.

[7] ——, "Multidimensional scaling of listener responses to synthetic speech," 2005.

[8] H. Kawahara, I. Masuda-Katsuse, and A. de Cheveigné, "Restructuring speech representations using a pitch-adaptive time–frequency smoothing and an instantaneous-frequency-based F0 extraction: Possible role of a repetitive structure in sounds," *Speech communication*, vol. 27, no. 3, pp. 187–207, 1999.

[9] C. Liu and D. Kewley-Port, "STRAIGHT: A new speech synthesizer for vowel formant discrimination," *Acoustics Research Letters Online*, vol. 5, p. 31, 2004.

[10] S. Imai, T. Kobayashi, K. Tokuda, T. Masuko, K. Koishida, S. Sako, and H. Zen, "Speech signal processing toolkit (SPTK), version 3.6," 2012.

[11] K. Tokuda, Y. Nankaku, T. Toda, H. Zen, J. Yamagishi, and K. Oura, "Speech synthesis based on Hidden Markov Models," *Proceedings of the IEEE*, vol. 101, no. 5, pp. 1234–1252, 2013.

[12] T. Yoshimura, "Simultaneous modeling of phonetic and prosodic parameters, and characteristic conversion for HMM-based Text-To-Speech systems," Ph.D. dissertation, Ph. D. thesis, Nagoya Institute of Technology, 2002.

[13] T. Tomoki and K. Tokuda, "A speech parameter generation algorithm considering global variance for HMM-based speech synthesis," *IEICE TRANSACTIONS on Information and Systems*, vol. 90, no. 5, pp. 816–824, 2007.

[14] IEEE, "IEEE recommended practice for speech quality measurement," vol. 17, no. 3, pp. 225 – 246, sep 1969.

[15] M. Cooke, C. Mayo, and C. Valentini-Botinhao, "Intelligibility-enhancing speech modifications: the Hurricane Challenge," in *Proc. Interspeech*, Lyon, France, 2013.

[16] K. Richmond, P. Hoole, and S. King, "Announcing the electromagnetic articulography (day 1) subset of the mngu0 articulatory corpus," in *Proc. Interspeech*, Florence, Italy, August 2011, pp. 1505–1508.

[17] I. Borg and P. J. Groenen, *Modern Multidimensional Scaling*. Springer, 2005.

Prosodic analysis of storytelling discourse modes and narrative situations oriented to Text-to-Speech synthesis

Raúl Montaño[1], *Francesc Alías*[1], *Josep Ferrer*

[1] Grup de Recerca en Tecnologies Mèdia, La Salle, Universitat Ramon Llull, Barcelona
{raulma;falias;st15228}@salle.url.edu

Abstract

The generation of synthetic speech with a certain degree of expressiveness has been successful for some particular applications or speaking styles (e.g. emotions). In this context, there is a particular speaking style with subtle speech nuances that may be of great interest for delivering expressive speech: the storytelling style. The purpose of this paper is to define a first step towards developing a storytelling Text-to-Speech (TTS) synthesis system by means of modelling the specific prosodic patterns (pitch, intensity and tempo) of this speaking style. We base our analysis of a tale in Spanish on discourse modes present in storytelling: narrative, descriptive and dialogue. Moreover, we introduce narrative situations (neutral narrative, post-character, suspense and affective situations) within the narrative mode, which are analysed at the sentence level. After grouping the sentences into modes and narrative situations, we analyse their corresponding prosodic patterns both objectively (via statistical tests) and subjectively (via perceptual test considering resynthesized sentences). The results show that the statistically validated prosodic rules perform equally (or even better) than the original prosody in most sentences.

Index Terms: storytelling, prosodic analysis, narrative situations, TTS, Harmonic plus Noise Model

1. Introduction

Storytelling speaking style has been studied following quite different approaches for the analysis of the specific characteristics of stories and tales. For example, in [1] the authors analysed storytelling according to a common structure of tales (title, exposition, triggering event, a series of scenes, ending and epilogue), whereas in [2] the authors divided the tale into three discourse modes [3] (narrative, descriptive and dialogue), concluding that the storyteller prosody is influenced by discourse modes. In [4], tales and news reading styles were prosodically analysed and compared showing prosodic differences between both styles. In contrast to these global approaches, other works addressed the tale analysis at the sentence level. Specific narrative passages (global storytelling speaking style, increasing suspense and sudden suspense) were studied, modelled and synthesized in [5]. On the other hand, some works modelled the whole story following an emotional approach, for both analysis [6] [7] [8] and synthesis purposes [9] [10], while others only considered emotions for the characters of the story [11].

Nevertheless, none of these works offers a complete solution to deal with the prosodic analysis and modelling of the storytelling speaking style at the sentence level oriented to speech synthesis of all the expressive registers of a storyteller. The storyteller is the person narrating the tale, e.g., the story and the situations that the characters are experiencing. Optionally, he/she can interpret all/part of the characters turns too. To that effect, storytellers make use of a wide range of speech variability in order to convey the necessary expressiveness to capture the audience's (generally, children) attention. For example, they may use rhythm changes or include pauses of different duration, add suspense to the voice, use much more variation of pitch and intensity than other speaking styles such as the newsreader speaking style, stretch some words, etc. [5]. Dialogues are also present in many novels and tales because it is a factor that can engage the audience in the story in a greater way as they can read/listen to the characters directly. Moreover, in oral communication the narrator may give different voices and emotional content to different characters to enhance realism and entertainment. In contrast, in the narrative and descriptive modes, more subtle nuances appear to convey the storytelling style.

In this work, we propose a first approach to cope with that issue performing a prosodic analysis of a story narrated by a Spanish storyteller based on storytelling discourse modes [2] [3]. However, we consider that the prosodic analysis should be conducted at the sentence level to capture all the potential expressive registers of a storyteller following a bottom-up approach. To that effect, we introduce new sub-modes (narrative situations) inside the narrative mode to cope with the sentence level analysis, which will be the basis for further synthesis purposes. Finally, we have chosen the main character of the story to analyse the dialogue mode using an annotation scheme based on basic emotions. After a two-phase analysis of the story at hand, the narrative situations are both objectively and subjectively validated by means of statistical significance analysis and a subsequent preliminary perceptual test considering the corresponding synthesis from the extracted prosodic rules. For the dialogue mode, we compare our results with other studies that have analysed basic emotions to observe if emotions in storytelling show equal or specific prosodic patterns. The obtained emotional rules are also tested in the synthesis phase.

This paper is structured as follows. Section 2 reviews related work on the analysis and synthesis of storytelling speaking style. In Section 3, the proposed approach for storytelling speech analysis is described. Next, the prosodic analysis is detailed in Section 4. Then, the perceptual evaluation with synthesis using the extracted prosodic rules is described in Section 5. Finally, some conclusions and future work are present in Section 6.

2. Related Work

The particular challenges of generating storytelling speaking style were discussed in [9], where the authors stumbled upon this problem as the Text-To-Speech (TTS) system of their embodied digital storyteller did not offered the desired expressive-

ness. According to the authors, the lack of flexibility of the considered TTS system was the main problem. Probably, the fact that the prosodic model was based on emotion profiles borrowed from the literature was also a relevant factor, since they are not entirely well-suited for recreating the storytelling speaking style (e.g., different approaches like [5] seemed to obtain better synthetic results). A later work by some of the authors (centred on interactive storytelling) also remarked that a main obstacle in their work was the synthetic quality of their TTS system [12]. In [13], similar conclusions were obtained on a project devoted to give a robot the ability to tell tales to children. The authors claimed that in storytelling there are particular expressive turns, such as different degrees of emphasis, changes of registers and tempo, different characters, etc., that must be included in the synthetic discourse.

As in [9], later works have linked basic emotions with storytelling. Emotional tags were used to analyze a storytelling speech corpus in [6], which led to a certain degree of correspondence with previously reported emotional acoustic profiles in the literature. Nevertheless, some particular contradictory results (as pitch decrease for anger) were also obtained. Moreover, emotional acoustic models borrowed from the literature were only used for characters from stories in [11]. Although the model was preliminary and needed further work, the synthetic results showed that the changed emotional fragments compared to the neutral fragments were mostly noticeably different, and five emotions were accepted at a reasonable rate.

In [5], the authors only modelled global storytelling speaking style and suspense situations (increasing suspense and sudden suspense). The resynthesized speech generated according to the obtained set of prosodic rules obtained good synthetic quality. However, the rules where highly preliminary because of the very small amount of data considered for the analysis (2 sentences for the sudden suspense and 1 sentence for the increasing suspense). Although the authors proposed a 'global storytelling style', we consider that there is still room for further research towards defining a truly general storytelling style.

A high level annotation scheme according to a common structure of tales (title, exposition, triggering event, a series of scenes, refrain and epilogue [14]), was used in [1] to analyse the prosody of the aforementioned tale sections. However, the authors pointed out that an annotation of affect and emotional tags at the sentence level would be necessary to refine their results. Furthermore, a high level prosodic analysis of a tale was also carried out in [2] in order to perform automatic classification of sentences. The authors labelled the text of a tale among narrative mode, descriptive mode and dialogue mode. The authors argued that prosody is used to mark discourse modes.

Taking these works into account, we base our analysis on storytelling discourse modes but going into the sentence level considering our final synthesis goal. Therefore, new sub-modes, denoted as narrative situations, have been defined as explained in Section 3. Then, a series of prosodic rules are extracted and validated (see Section 4).

3. Narrative situations and character emotions

How should one deal with the classification of text and expressive content in storytelling? Trying to categorize each sentence of a story into one specific basic or secondary emotion or attitude does not seem to be a very good idea. First, relating the narrative style to emotions seems inappropriate, as the narrator is not self-experiencing the emotions and it is not his/her intention to simulate them but to engage the audience in the story. Secondly, gathering a representative corpus for each emotional or attitude to look for speech correlates would be thoroughly intractable [15].

The annotation framework that is used in this work for the further generation of synthetic storytelling speech is based on discourse modes [2] [3]. Specifically, among all discourse modes (narrative, descriptive, argumentative, explanatory and dialogue), the fiction literature (storytelling) typically contains the narrative mode, the descriptive mode and the dialogue mode [2] [3]. In the literary field, the narrative mode is mainly used to inform the listener/reader about the actions that are taking place and affect the characters of the story. Therefore, this mode includes a great amount of text that a storyteller (an expressive one at least) conveys in different expressive registers typically at the sentence level.

The story analysed in this work is "Harry Potter and the Philosopher's Stone" read by a Spanish male storyteller[1]. For the indirect discourse, we have analysed the first chapter of the story, whereas for the study of the dialogue mode, we consider the interventions of the main character of the story (Harry Potter) extracted from the whole story. We followed a two-phase analysis method: a linguistic analysis and a subsequent perceptual refinement. The annotation at the sentence level of the text of the story was entrusted to two experts on text classification. They were instructed to classify sentences as descriptive mode, dialogue mode (Harry's interventions) and narrative mode (see phase 1 of Figure 1).

For the narrative mode they were instructed to classify the sentences according to valence sub-modes (neutral, positive and negative sentences), since it is a useful representation for affective situations where the emotional state is not fully defined [16]. They also were asked to classify what we call post-character sentences. These situations correspond to sentences of indirect discourse immediately following a direct discourse (character intervention) with usually a declarative verb on the third person [17]. Sentences where the annotators did not agree (9.1% of the total number of sentences) were discarded for the second phase.

Once the sentences were classified from text, they were presented to two experts on speech technologies to further analysis (see phase 2 of Figure 1). A briefing with some examples of the different categories they had to listen was given beforehand. Since we are interested in modelling the prosody of the narrator, we consider that we cannot limit our annotation scheme to text and structure characteristics of stories and tales. Their observations were that the affective situations (sentences with positive and negative valence) were too heterogeneous perceptually, and needed a refinement based on activation. Therefore, they classified the affective sentences into Positive/Active, Positive/Passive, Negative/Active and Negative/Passive situations. In addition, they noticed that several neutral and negative sentences possessed a certain degree of suspense. These sentences seemed to possess a greater suspense and tension (expressed with a softer voice) typically caused by a strange event or something the characters of the story are unaware of, leaving the audience to think that something important may happen soon. After considering this fact, we decided to take them into account in the acoustic analysis as a new category: suspense situations. With respect to phase 1, 79% of the suspense sentences came

[1] http://www.ivoox.com/podcast-harry-potter-piedra-filosofal-j-k-rowling_sq_f137546_1.html

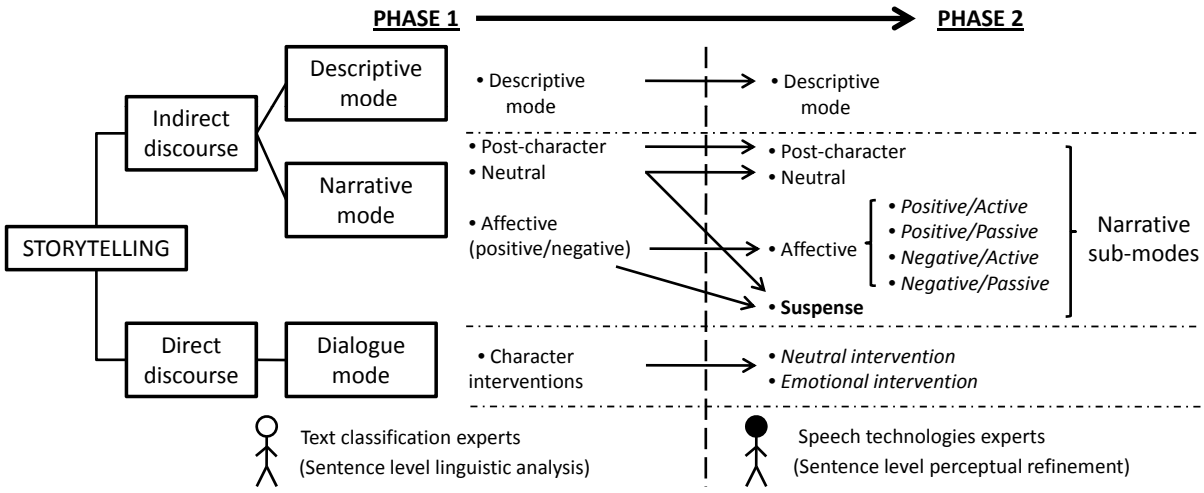


Figure 1: *Diagram representing the followed approach in the present work to analyse the storytelling corpus. Categories in phase 2 are in italics if they were refined from previous categories, in bold if they resulted in new categories and in standard style if they were the same from phase 1.*

Table 1: *Total amount of identified sentences in the speech corpus. Time is expressed as mm:ss.*

Category	# sentences	time
Neutral narrative	46	03:15
Negative/Passive	36	02:34
Negative/Active	30	02:20
Positive/Passive	30	02:22
Positive/Active	31	02:56
Post-character	30	01:03
Suspense	27	01:54
Descriptive mode	30	03:11
TOTAL	260	19:35

(a) Narrative & Descriptive modes

Category	# sentences	time
Neutral	14	00:33
Hot anger	18	00:35
Cold anger	15	00:30
Joy	14	00:15
Sadness	15	00:36
Surprise	12	00:19
Fear	18	00:56
TOTAL	106	03:44

(b) Dialogue mode

from the neutral sentences whereas 21% came from negative sentences (see transition from phase 1 to phase 2 in Figure1). In the related literature, two types of suspense situations (sudden and increasing suspense) present in storytelling have already been modelled [5]. Although some sentences could be related to those kinds of suspense, it was such a small speech corpus that led us to omit these subdivision for the following prosodic analysis.

For the classification of the dialogue mode we use a basic emotions annotation scheme. If a narrator interprets the characters, he/she typically modifies his/her voice into a more exaggerated register of expressions, where full-blown emotions may be manifested [11]. From the gathered corpus, both experts on speech technologies were asked to classify the sentences into six basic emotions (hot anger, cold anger, joy, sadness, surprise and fear) besides a neutral category.

The final collected corpus for the prosodic analysis is detailed in Table 1. We are aware that this is not a very extensive corpus, but it serves as a preliminary step to observe the viability of the approach.

4. Prosodic analysis

In this section, we present the results of the prosodic analysis performed on the sentences collected and labelled after the two-phase process described in Figure 1. We analyse pitch, intensity and tempo. The parameter for modelling tempo is speaking rate (SR) in syllables per second without pauses. Pitch is represented with mean pitch (MP) and pitch standard deviation (PSD) in Hertz while for intensity, we extracted mean intensity (MI) in decibels. We used the speech analysis software Praat [18] to extract these prosodic parameters. In order to maximize the measurement precision of pitch, each sentence received optimal pitch floor and ceiling values computed with the MOMEL plug-in for Praat [19] (manually corrected if needed). On the other hand, the SR was measured with the ADoTeVA Praat plug-in[2]. The segmentation of the speech corpus into words, syllables and phonemes was carried out with the EasyAlign tool [20], and was manually corrected afterwards.

First, we analyze the indirect discourse, i.e., the narrative mode with its associated narrative situations and the descriptive mode. We select the neutral narrative situation as the reference category, so the results of the rest of categories are referenced to this one in terms of relative percentage difference. An independent samples t-test has been performed using the SPSS software to check significant differences with respect to the neutral narrative situation and between all categories with pairwise comparisons. From this statistical analysis we want to determine to what extent categories are different and which parameters are more significant. The p-value used includes a correction when equal variances can not be assumed.

The dialogue mode analysis shows the prosodic results of the basic emotions present in the main character of the story (Harry Potter). We compare these results with other studies that have analyzed acted basic emotions in order to observe if storytelling emotions show equal or specific prosodic patterns.

Table 2: *Averaged results of the indirect discourse mode analysis. Statistical significance tests: * stands for p<0.05, while ** represents p<0.01. No * means no significant difference.*

Narrative mode	MP [Hz]	PSD [Hz]	SR [syll/sec]	MI [dB]
Neutral narrative	104.0	31.0	7.7	71.0
Post-character	-22.3%**	-50.2%**	-12.1%**	-4.3%**
Suspense	-5.7%**	-19.0%**	-10.9%**	-4.3%**
Negative/Passive	-12.9%**	-35.3%**	-8.0%**	-3.8%**
Negative/Active	+7.9%*	-3.3%	-2.2%	+1.2%
Positive/Passive	-8.9%**	-20.2%**	-8.4%**	-2.6%**
Positive/Active	+18.3%**	+24.3%**	-5.5%**	+2.5%**
Descriptive mode	+1.0%**	+10.0%**	-8.7%**	+0.3%

4.1. Indirect discourse results

Prosodic results for the narrative and descriptive modes are shown in Table 2, where they are referenced with respect to the neutral narrative situation. The statistical analysis is also depicted in Table 2 and the rest of statistical significance comparisons are shown in Table 3. The first global conclusion that arises from the results is that, in general, SR is not significantly different across categories. The analysed storyteller shows a fast SR for neutral narrative and Negative/Active situations, whereas in the rest of situations a slow SR is manifested, probably to allow the audience (children, in general) to follow the story.

Post-character sentences obtained the lowest averaged pitch and intensity values, which is in agreement with the perception of the speech technologies experts that, in general, these sentences sounded muffled. The SR has a mid-low value. The suspense situation shows low prosodic parameters too. As stated in [1], it seems necessary at least a low mean intensity to generate intimacy or suspense. These two situations only show significant differences in terms of pitch as it can be observed in row six of Table 3.

The results for the affective situations show that prosodic parameters from active sentences are significantly higher than the parameters from passive sentences with the exception of SR, which do not always follow this behaviour (see rows 16 to 19 in Table 3). These results quite agree with the established consensus in the literature that active sentences entail higher frequency, intensity and speaking rate [16]. Results in Table 2 show that sentences with positive evaluation have slightly higher prosodic values compared to sentences with different evaluation but with the same activation, with the exception of SR, which is lower. Although there are no clear acoustic correlations with valence in the literature, in [16], a higher mean frequency for a male voice was reported for positive valence, just as the results observed in the affective situations results of Table 2. It is worth pointing out that the passive categories have lower MP and MI when compared to the neutral narrative style, while for the active categories the opposite happens. On the other hand, SR for all the affective situations is slower than the neutral narrative situation SR. Finally, the PSD of Positive/Active sentences is the only one that surpasses the neutral narrative category.

Descriptive mode sentences have a higher MP and PSD than the neutral narrative situation whereas mean intensity is not significantly higher. The SR, however, is lower. All this information can be linked to what was perceived while listening to the speech corpus, as the narrator emphasizes certain adjectives and adverbs, which yields to greater pitch variability, and he stretches these words too in order to emphasize. The descriptive mode and the Positive/Active situation are the only categories that show no significant differences in their prosodic patterns (see row 14 in Table 3). One possible explanation is that the narrator when is describing tends to show a cheerful mood, but further investigation may disambiguate both categories.

[2] http://celinedelooze.com/MyHomePage/Praat.html

Table 3: *Results for the independent samples t-test analysis of indirect discourse categories: * stands for p<0.05, while ** represents p<0.01. P-C: Post-Character, PA: Positive/Active, PP: Positive/Passive, NA: Negative/Active, NP: Negative/Passive, SUS: Suspense, DM: Descriptive Mode.*

Compared categories	Test results			
	MP	PSD	SR	MI
P-C vs. DM	**	**	0.26	**
P-C vs. NA	**	**	**	**
P-C vs. PA	**	**	*	**
P-C vs. PP	**	**	0.27	0.13
P-C vs. NP	**	**	0.19	0.75
P-C vs. SUS	**	**	0.70	0.70
SUS vs. PA	**	**	*	**
SUS vs. NA	**	*	**	**
SUS vs. DM	**	**	0.41	**
SUS vs. PP	0.45	0.83	0.41	**
SUS vs. NP	**	**	0.30	0.34
PP vs. NP	**	**	0.99	0.06
PA vs. NA	**	**	0.18	**
PA vs. DM	0.55	0.97	0.20	0.20
NA vs. DM	**	**	**	**
PP vs. PA	**	**	0.33	**
PP vs. NA	**	*	*	**
NP vs. PA	**	**	0.35	**
NP vs. NA	**	**	*	**
DM vs. PP	**	**	0.91	**
DM vs. NP	**	**	0.79	**

Table 4: *Averaged results of the character emotions analysis.*

Emotion	MP [Hz]	PSD [Hz]	SR [syll/sec]	MI [dB]
Neutral	108.0	25.8	7.2	70.0
Hot anger	+82.8%	+112.3%	-20.6%	+9.1%
Cold anger	+42.4%	+69.0%	-16.7%	+4.0%
Joy	+28.9%	+67.6%	-11.2%	+7.2%
Sadness	-11.5%	-28.5%	-21.6%	-3.3%
Surprise	+45.2%	+92.7%	-15.9%	+0.7%
Fear	+29.1%	+27.2%	-2.2%	+5.3%

4.2. Direct discourse results

To analyse part of the dialogue mode present in storytelling we selected the main character (Harry Potter). The narrator interprets Harry without changing his voice too much, but it is noticeable he tries to imitate the voice of a pre-teenager.

Regarding emotional prosodic results (see Table 4), it is remarkable that all the emotions have a slower SR than the character neutral voice. In general, anger, joy, surprise and fear tend to have a faster SR in the literature [21] [22] [23] [24]. However, the difference between joy and happiness is not so clear in the literature. For example, In [22] the authors clearly

separated them and proposed a decrease in tempo for happiness (as in [24]) and an increase for joy. Results of SR from Harry's emotions are the ones which have more conflict when compared to the general literature focused on emotion analysis. As a preliminary conclusion, it seems that storytellers speak slower even in the character emotions (besides the indirect discourse). This can be due to the fact that they need to draw the audience attention and allow them to be able to follow all the delivered information. Thus, in this parameter may be the main difference with respect to more natural or spontaneous emotions.

Hot anger has the most exaggerated values of MP, PSD and MI of all the emotional catalogue. The raise of the mentioned prosodic parameters is quite coherent with previous studies focused on basic emotions [21] [22] [23]. Cold anger has the same changes as hot anger but not so wide. Joy shows the highest MI right after hot anger, and its pitch related values are quite high in general. Sadness is the emotion which has more relationship with the acoustic profiles reported in the literature, as it entails a decrease in all the prosodic parameters [21] [22] [23] [25]. Surprise, which is usually related to an increase of the prosodic parameters with respect to a neutral register, has also relationship with other studies (except for speaking rate as well) [23]. Fear has a relative coherency with the literature. In general, pitch, intensity and speaking rate also increase in fear when compared to a neutral register [22] [23]. From Table 4, we can see that MP, PSD and MI increase. The SR obtained for fear is the highest of all the emotions, almost the same as the one for Harry's neutral voice.

5. Speech synthesis evaluation

The main objective of the synthesis evaluation stage performed in this work is to subjectively validate the rules obtained for the different discourse modes (see Tables 2 and 4), as a complement of the objective statistical analysis performed in Section 4. We evaluate how the extracted prosodic rules perform against the original prosody of the sentences.

We applied these prosodic rules to a randomly selected set of sentences from the corpus at hand using a synthetic female voice obtained with the TTS synthesizer of La Salle R&D. We resynthesized 52 sentences (4 sentences for each category) with the obtained prosodic rules (PR) and the same 52 sentences applying the original prosody (OP) of each sentence. The modifications and final signal resynthesis were done using a MATLAB implementation of Harmonic plus Noise Models (HNM) [26]. In contrast to other implementations where the maximum voiced frequency is allowed to vary [27], the implementation used in this paper is fixed at 5Khz based on [28].

The synthetic results were evaluated using the online TRUE platform [29]. The subjective test was performed by 15 people, from which 9 are male and 6 female with a mean age of 34 (only 5 people are familiar with the field of speech technologies). The perceptual test is designed considering a 5-level CMOS scheme (OP much better, OP better, no difference, PR better and PR much better), and it is composed of 52 comparisons of the same sentence resynthesized with PR and OP, which are compared including the original sentence of the audio book as a reference.

Figure 2 shows the results obtained for the sentences belonging to the indirect discourse. As a general result, it can be observed that the most extreme cases of the 5-level CMOS range are the least chosen options, showing that both transformations (PR and OP) are perceived similarly. However, post-character sentences tend to be preferred when the original prosody is applied. This can be due to the fact that sometimes

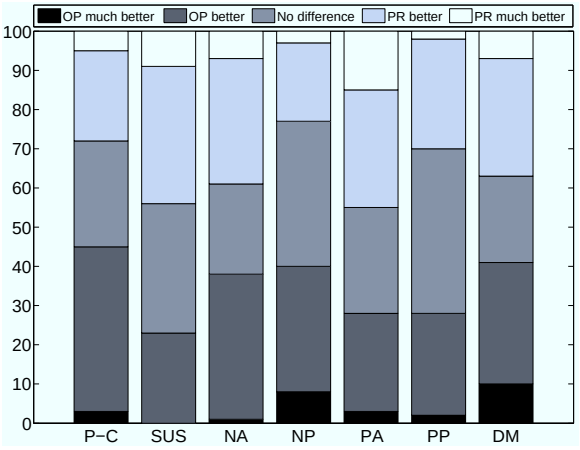


Figure 2: *Percentages bars of the results from the indirect discourse synthesis evaluation.*

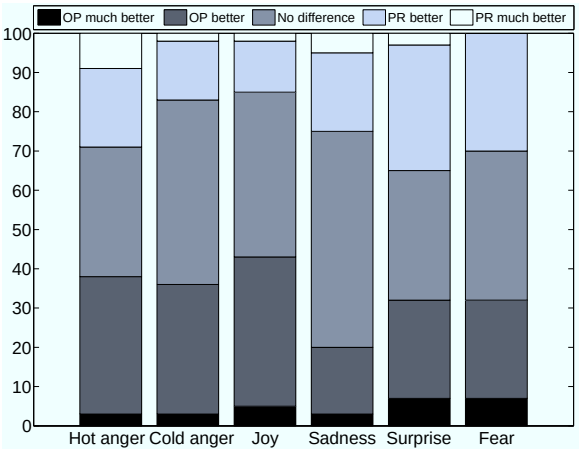


Figure 3: *Percentages bars of the results from the direct discourse synthesis evaluation.*

the narrator maintains emotional traces from the previous character intervention whereas in other post-character sentences he barely is expressive. Suspense sentences have obtained very good results when synthesized with the PR.

The results extracted from the analysis of the emotional sentences of the main character show that most evaluators did not notice clear differences between both prosodic options (see Figure 3), which is a positive result in terms of extracting preliminary general prosodic patterns. However, the PR are not as clearly preferred to the OP (in hot anger, cold anger and joy above all) as in the indirect discourse. This is due to the fact that the emotions show greater prosodic variability and the emotional speech corpus gathered could not be extended.

6. Conclusions

In this paper, we have presented a first approach to cope with the prosodic analysis and modelling of the subtle expressive registers present in the storytelling speaking style at the sentence level. After a linguistic and perceptual analysis of a story based on storytelling discourse modes (narrative, descriptive and dialogue) we have introduced some narrative situations. Next, we have performed a prosodic analysis and extracted preliminary

prosodic rules that have been implemented in a HNM synthesis phase. The outcome of the statistical and synthesis evaluation stages show a first confirmation that there are expressive categories inside the storytelling speaking style that show specific prosodic cues and can be modelled for synthesis purposes. This confirms and extends the conclusions in [5], where specific suspense situations were modelled giving room for further investigation of storytelling expressive registers.

These results encourage us to follow this approach in further studies to look for a *truly* generalizable storytelling speaking style prosodic model. We plan to include more narrators or the same narrator telling a similar story, cross-language analysis, or other forms of stories such as short fairy tales. Short fairy tales tend to have a more common structure than novels, so it would be interesting to observe how the narrative situations are mapped in such a structure. Regarding the used synthesis method, we consider that HNM-based TTS synthesis is a good approach to address storytelling speech thanks to the synthesis flexibility it allows. Nonetheless, other methods like concatenative synthesis can also be considered and compared.

7. Acknowledgements

The first author of this paper would like to acknowledge the support of the Catalan Government (SUR/ECO) for the predoctoral FI grant No. 2013FI_N 00790. We also thank Àngel Calzada and Dr. Joan Claudi Socoró for their support in the HNM synthesis implementation.

8. References

[1] D. Doukhan, A. Rilliard, S. Rosset, M. Adda-Decker, and C. d'Alessandro, "Prosodic analysis of a corpus of tales," in *Interspeech*, 2011, pp. 3129–3132.

[2] J. Adell, A. Bonafonte, and D. Escudero, "Analysis of prosodic features towards modelling of emotional and pragmatic attributes of speech," *Procesamiento del lenguaje natural*, no. 35, pp. 277–283, 2005.

[3] H. Blancafort, A. Tusón, and A. Valls, *Las Cosas del decir: manual de análisis del discurso*, ser. Ariel Letras. Editorial Ariel, 2007.

[4] O. Jokisch, H. Kruschke, and R. Hoffmann, "Prosodic reading style simulation for Text-to-Speech synthesis," in *Affective Computing and Intelligent Interaction*, ser. Lecture Notes in Computer Science, J. Tao, T. Tan, and R. W. Picard, Eds. Springer Berlin Heidelberg, 2005, vol. 3784, pp. 426–432.

[5] M. Theune, K. Meijs, D. Heylen, and R. Ordelman, "Generating expressive speech for storytelling applications," *IEEE Trans. on Audio, Speech and Language Processing*, pp. 1137–1144, 2006.

[6] C. O. Alm and R. Sproat, "Perceptions of emotions in expressive storytelling," in *Interspeech*, 2005, pp. 533–536.

[7] C. O. Alm, D. Roth, and R. Sproat, "Emotions from text: machine learning for text-based emotion prediction," in *Proc. of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, Vancouver, B. C., Canada, 2005, pp. 579–586.

[8] V. Francisco, P. Gervás, M. González, and C. León, "Expressive synthesis of read aloud tales," in *Artificial and Ambient Intelligence*, 2007, pp. 179–186.

[9] A. Silva, M. Vala, and A. Paiva, "The storyteller: Building a synthetic character that tells stories," in *Proc. of the Workshop Multimodal Communication and Context in Embodied Agents*, 2001, pp. 53–58.

[10] F. Burkhardt, "An affective spoken storyteller," in *Interspeech*, 2011, pp. 3305–3306.

[11] H. Buurman, "Virtual storytelling: Emotions for the narrator," Master's thesis, University of Twente, The Netherlands, 2007.

[12] A. Silva, G. Raimundo, A. Paiva, and C. Melo, "To tell or not to tell...Building an interactive virtual storyteller," in *Proc. of the Language, Speech and Gesture for Expressive Characters Symposium, Artificial Intelligence and the Simulation of Behaviour Covention*, March 2004.

[13] R. Gelin, C. d'Alessandro, O. Deroo, Q. A. Le, D. Doukhan, J.-C. Martin, C. Pelachaud, A. Rilliard, and S. Rosset, "Towards a storytelling humanoid robot," *AAAI Fall Symposium Series on Dialog with Robots*, pp. 137–138, 2010.

[14] V. A. Propp, *Morphology of the Folktale*, 2nd ed., ser. Publications of the American Folklore Society. University of Texas Press, 1968.

[15] R. Cowie, "Describing the emotional states expressed in speech," in *ISCA Workshop on Speech & Emotion*, Northern Ireland, 2000, pp. 11–18.

[16] M. Schröder, "Speech and emotion research: An overview of research frameworks and a dimensional approach to emotional speech synthesis," Ph.D. dissertation, Saarland University, 2004.

[17] N. Mamede and P. Chaleira, "Character identification in children stories," in *Advances in Natural Language Processing*, ser. Lecture Notes in Computer Science, J. L. Vicedo, P. Martínez-Barco, R. Muñoz, and M. Saiz Noeda, Eds. Springer Berlin Heidelberg, 2004, vol. 3230, pp. 82–90.

[18] P. Boersma and D. Weenink, "Praat: doing phonetics by computer [computer program]. (v.5.3.39)," retrieved 6 January 2013 from http://www.praat.org/.

[19] D. Hirst, "A Praat plugin for Momel and INTSINT with improved algorithms for modelling and coding of intonation," in *Proc. of the 16th international congress of phonetic sciences*, 2007, pp. 1233–1236.

[20] J. P. Goldman, "EasyAlign: An automatic phonetic alignment tool under Praat." in *Interspeech*, 2011, pp. 3233–3236.

[21] I. Iriondo, F. Alías, J. Melenchón, and M. A. Llorca, "Modeling and synthesizing emotional speech for Catalan Text-to-Speech synthesis," in *Tutorial and Research Workshop on Affective Dialog Systems*, 2004, pp. 197–208.

[22] F. Burkhardt and W. Sendlmeier, "Verification of acoustical correlates of emotional speech using formant-synthesis," in *Proc. of the ISCA Workshop on Speech and Emotion*, 2000, pp. 151–156.

[23] I. Iriondo, R. Guaus, A. Rodríguez, P. Lázaro, N. Montoya, J. M. Blanco, D. Bernadas, J. M. Oliver, D. Tena, and L. Longhi, "Validation of an acoustical modelling of emotional expression in Spanish using speech synthesis techniques," in *Proc. of the ISCA Workshop on Speech and Emotion*, 2000, pp. 161–166.

[24] M. Kienast, A. Paeschke, and W. F. Sendlmeier, "Articulatory reduction in emotional speech," in *EUROSPEECH*, 1999, pp. 117–120.

[25] R. Banse and K. R. Scherer, "Acoustic profiles in vocal emotion expression." *J. Pers. Soc. Psychol.*, vol. 70, no. 3, pp. 614–636, Mar. 1996.

[26] À. Calzada and J. C. Socoró, "Voice quality modification using a Harmonics Plus Noise Model," *Cognitive Computation*, pp. 1–10, 2012.

[27] Y. Stylianou, "Harmonic plus Noise Models for speech, combined with statisticals methods, for speech and speaker modification," Ph.D. dissertation, École Nationale Supérieure des Télécommunications, 1996.

[28] D. Erro, "Intra-lingual and cross-lingual voice conversion using Harmonic plus Stochastic Models," Ph.D. dissertation, Technical University of Catalonia, 2008.

[29] S. Planet, I. Iriondo, E. Martínez, and J. A. Montero, "TRUE: an online testing platform for multimedia evaluation," in *Proceedings of the Second International Workshop on EMOTION: Corpora for Research on Emotion and Affect at the 6th Conference on Language Resources & Evaluation*, ser. LREC '08, Marrakech, Morocco, 2008.

Objective evaluation measures for speaker-adaptive HMM-TTS systems

Ulpu Remes, Reima Karhila, Mikko Kurimo

Department of Signal Processing and Acoustics, Aalto University School of Electrical Engineering, Finland

firstname.lastname@aalto.fi

Abstract

This paper investigates using objective quality measures to evaluate speaker adaptation performance in HMM-based speech synthesis. We compare several objective measures to subjective evaluation results from our earlier work about 1) comparison of speaker adaptation methods for child voices and 2) effects of noise in speaker adaptation. The results analysed in this work indicate a reasonable correlation between several objective and subjective quality measures.

Index Terms: adaptation, speech synthesis, evaluation

1. Introduction

Hidden Markov model (HMM) based text-to-speech (TTS) framework [1] is an attractive alternative to conventional concatenative speech synthesis. While concatenative systems typically produce natural and understandable speech, HMM-TTS systems are more flexible and can be adapted to mimic different speaking styles or speakers based on a limited amount of adaptation data [2].

Speaker adaptation performance in HMM-TTS systems is typically evaluated using subjective listening tests [3]. The samples generated with speaker-adapted models are rated based on whether the synthesised voice sounds like the target speaker and on the perceived naturalness. While listening tests are necessary to confirm how differences between adaptation methods are perceived and appreciated, subjective evaluation is not an efficient tool for tasks such as parameter tuning that require iterative evaluation.

Numerous objective quality measures have been developed for speech quality evaluation in telecommunication systems [4, 5]. In addition to telecommunication systems, the measures have been used to evaluate speech enhancement systems and have correlated well with subjective evaluations [6]. While the degradation introduced in speech transmission or enhancement may have a fundamentally different nature compared to samples generated with statistical speech synthesis [7, 8], the same objective measures have been applied to evaluate speech synthesis systems [9, 8, 10, 11].

In this work, we replicate and expand the previous studies [6, 10, 11] on correlation between objective quality measures and subjective listening test results. We focus on subjective evaluations of speaker-adaptation performance in HMM-TTS systems, using limited sets of data from our earlier works [12, 11]. Beside comparing objective measures, we investigate, for the spectrum-based measures, whether measuring the spectrum as synthesised or analysed after a complete waveform synthesis affects the measures.

The rest of the paper is organised as follows. In Section 2, we describe the objective evaluation measures used in this work. In Section 3, we revisit the listening test results used in previous research and evaluate the correlation between objective and subjective evaluations. All the speech synthesis systems represented in the results are personalised HMM-TTS systems. The results are discussed in Section 4, and Section 5 concludes the work.

2. Methods

2.1. Objective measures

HMM-based speech synthesis systems use statistical models to generate the spectral envelope and F_0 contour that are input to a vocoder that synthesises the final output waveform. To evaluate the system performance, the estimated parameters can be compared with equivalent parameters extracted from a reference speech sample [13, 2]. The mel-cepstral distance (MCD) is calculated as

$$MCD = \frac{1}{M} \sum_m \sqrt{2 \sum_d (c(d,m) - \hat{c}(d,m))^2} \qquad (1)$$

where $\hat{c}(d,m)$ and $c(d,m)$ denote the dth mel-cepstral coefficient of the test and reference signals in time frame m and M denotes the number of frames. We note that a synthesised test sample can be represented with the internal mel-cepstrum which is used as a synthesis parameter or with mel-cepstral coefficients extracted from the synthesised output. We compute and evaluate both internal and output mel-cepstral distances.

The other evaluation measures used in this work have been developed to assess the speech enhancement or transmission qualities. We focus on the frequency-weighted segmental SNR (FWS) [4] that exhibited a performance close standardised PESQ objective evaluation measure in a speech enhancement evaluation task [6]. We calculate the frequency-weighted segmental SNR in mel-spectral domain as

$$FWS = \frac{1}{M} \sum_m \sum_k W(k,m) \log_{10} \frac{X(k,m)^2}{(X(k,m) - \hat{X}(k,m))^2} \qquad (2)$$

where $\hat{X}(k,m)$ and $X(k,m)$ denote the kth mel-spectral component of the test and reference samples in time frame m. As proposed in [6], the mel-spectrum in each time frame m is normalised to unit area ($\sum_k X(k,m) = 1$) and the channels are weighted as

$$W(k,m) = X(k,m)^\gamma / \sum_k X(k,m)^\gamma, \qquad (3)$$

where $\gamma = 0.2$. The mel-spectral features $\hat{X}(k,m)$ that represent a synthesised test sample are calculated based on the internal mel-cepstral representation or extracted from the synthe-

sised output as discussed in Section 2.2. The estimated signal-to-noise ratio in each time frame is bound to $[0, 35]$ dB range as proposed in [14].

We additionally calculate the cepstral distance (CEP), log-likelihood ratio (LLR) [15], and weighted spectral slope (WSS) measure [16] using the implementations in COLEA toolbox [17]. The measures calculated with COLEA are computed based on the reference sample and synthesised output samples in time-domain. LLR is calculated based on order 10 linear prediction models.

2.2. Feature extraction

To compare the reference samples to samples generated with the HMM-TTS system, the synthesised samples were generated based on the phone alignment of the reference sample. The synthesised samples were associated with the internal mel-cepstral and spectral representation that correspond to the mel-cepstral and spectral features generated with the STRAIGHT vocoder [18]. MCD and FWS calculated based on the internal representations and STRAIGHT-based representations generated from the synthesised output are compared in this work. We additionally calculate the FWS measure based on FFT spectra calculated for the test and reference samples to compare the FFT and STRAIGHT spectrum.

The test and reference samples have 16 kHz sampling rate. The samples were processed in 25 ms Hamming windows with 5 ms shift between adjacent frames. The mel-filterbank applied to FFT or STRAIGHT spectra was estimated with VOICEBOX [19].

The objective measures were calculated based on 2 second samples extracted from the middle of the utterance as proposed in [11]. For some reason, the synthesis system used in this work occasionally introduces excess frames in the beginning or end of the sample. The comparison between the reference and test samples was therefore done at several frame delays $[-10 \ldots 10]$ and the best match was recorded.

3. Evaluation

3.1. Subjective test data

The objective measures are compared with mean opinion scores (MOS) collected in two speaker-adaptive HMM-TTS evaluations [12, 11]. Finnish speech data was used in both evaluations. The synthesis systems used in one evaluation share a common framework: the same prosody-prediction and model selection front-end are used and the phoneme sets are identical. The system parameters and training are described in the original papers [12, 11].

The first evaluation compared speaker adaptation methods for child voices. The differences between the synthesis systems in this evaluation stem from differences in the average voice training database and adaptation procedures [12]. Adaptation performance was evaluated in a subjective listening test where 26 listeners rated test samples from three target speakers based on similarity with the target speaker and naturalness. The samples were evaluated on a scale of 1–5. The mean opinion scores are reported in Figure 1.

The second evaluation studied using noisy and enhanced speech data for speaker adaptation in HMM-TTS systems. The differences between the synthesis systems stem from the differences in the adaptation training data that was either clean, noise-corrupted, or enhanced [11]. The evaluation focussed on the mel-cepstrum and excitation components, which were generated with the HMM-TTS system, whereas the F0 contours for synthesis were extracted from the reference samples. Adaptation to one female and one male speaker was evaluated, and the listening test samples included also noise-corrupted and enhanced natural speech samples. 26 listeners evaluated the samples based on their naturalness, similarity, and background intrusiveness as proposed in [11]. The subjective evaluation scales are described Table 1 and the mean opinion scores reported in Figure 2.

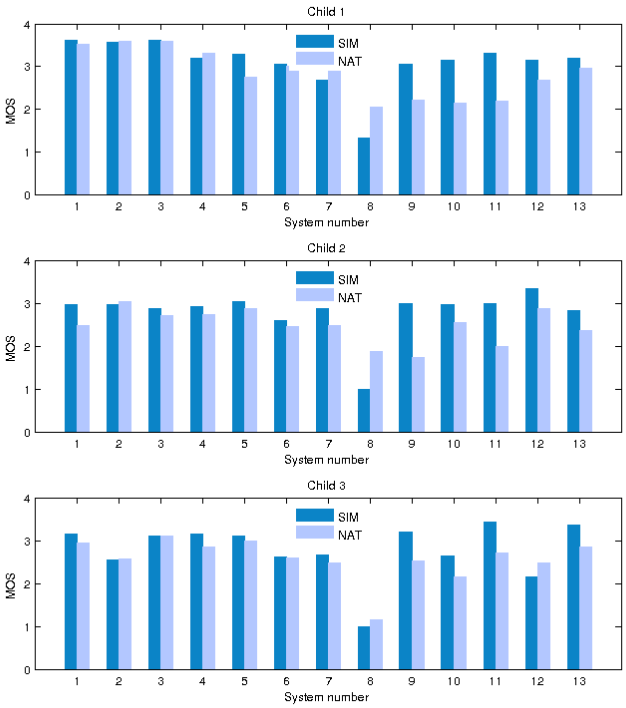


Figure 1: Mean opinion scores (MOS) on synthesised child speech. Thirteen systems were evaluated for similarity (SIM) and naturalness (NAT).

3.2. Evaluation measures

Concordance between the subjective and objective measures is assessed with a sample correlation coefficient $|\bar{r}|$. The standard sample correlation r is modified to marginalise the level differences between scores assigned to individual speakers n and emphasise the comparison between the tested conditions or systems. The modified sample correlation coefficient is calculated as

$$\bar{r} = \frac{1}{N} \sum_{n} r(n) \tag{4}$$

where N denotes the number of speakers and $r(n)$ is the speaker-conditioned sample correlation. The sample correlation between the subjective and objective scores assigned to speaker n is calculated as

$$r(n) = \frac{\sum_i (S_i(n) - \bar{S}(n))(O_i(n) - \bar{O}(n))}{\sqrt{\sum_i (S_i(n) - \bar{S}(n))^2 \sum_i (O_i(n) - \bar{O}(n))^2}} \tag{5}$$

Table 1: Subjective listening test scales

	Similarity (SIM)
5	Exactly like the same person
4	Quite like the same person
3	Somewhat different but recognisable as the same person
2	Quite like a different person
1	Like a totally different person

	Naturalness (NAT)
5	Completely natural
4	Quite natural
3	Somewhat unnatural but acceptable
2	Quite unnatural
1	Completely unnatural

	Background (BAK)
5	Clean
4	Quite clean
3	Somewhat noisy but not intrusive
2	Quite noisy and somewhat intrusive
1	Very noisy and very intrusive

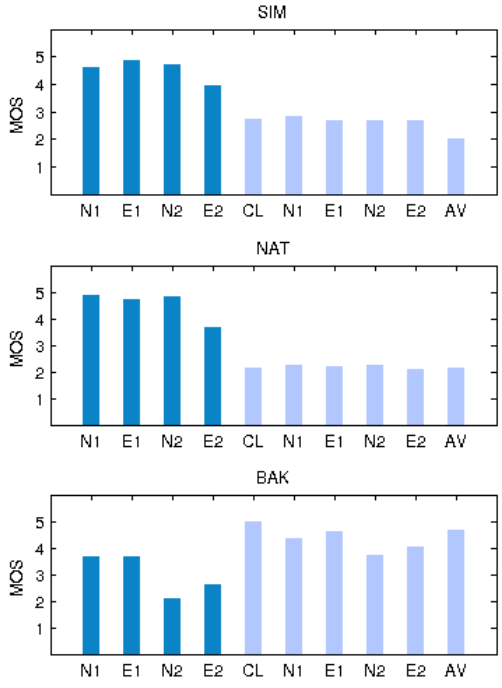


Figure 2: Mean opinion scores (MOS) on natural speech samples (dark colour) that have been corrupted with noise (N1–N2) and enhanced (E1–E2) and synthesised samples generated with HMM-based TTS. The synthesised samples represent the average voice model (AV) and models adapted with clean (CL) and noise-corrupted and enhanced data.

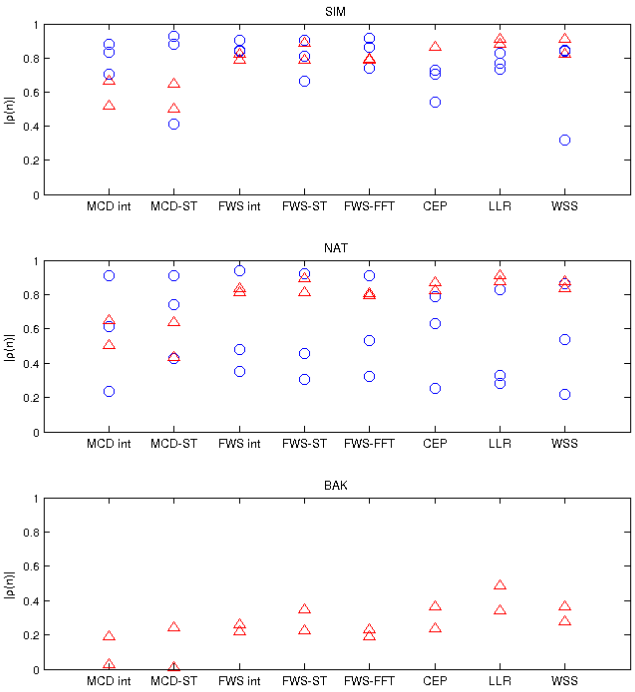


Figure 3: Linear correlation coefficients $|\rho(n)|$ between the objective evaluation measures and the subjective mean opinion scores. Results related to the first test with three child voices are indicated with circles and results related to the second test with one male and one female voice with are indicated with triangles.

where $S_i(n)$ and $O_i(n)$ denote the subjective and objective score calculated for test sample (i, n) and $\bar{S}(n)$ and $\bar{O}(n)$ the mean of the subjective and objective scores assigned to speaker n, $\bar{S}(n) = \sum_i S_i(n)$ and $\bar{O}(n) = \sum_i O_i(n)$. The subjective scores $S_i(n)$ are average scores calculated across the individual listener ratings for test sample (i, n). Listener ratings that deviated more than two standard deviations from the test sample mean were discarded as outliers and are not reflected in the averages $S_i(n)$.

3.3. Results

The sample correlations $|r(n)|$ between the objective measures and subjective listening test results are reported in Figure 3. We have evaluated the correlation between subjective evaluations and (a) MCD calculated based on the internal STRAIGHT representation and (b) STRAIGHT representation extracted from the synthesised output, (c) FWS calculated based on the internal representation, (d) representation extracted from the synthesised output and (e) FFT spectrum calculated from the synthesised output, (f) cepstral distance, (g) LLR and (h) WSS. FWS measures were calculated based on $K = 13$ and $K = 21$ mel-channels, but the differences in the results and in their correlation with MOS scores were small. The results reported in Figure 3 pertain to FWS measures calculated based on $K = 21$ mel-channels.

The sample correlations between the objective measures and subjective listening test results on the synthesised child voices are indicated with circles in Figure 3. We note that the subjective SIM and NAT evaluations are inter-dependent to certain extent, but their relationship is not linear ($|\bar{r}| = 0.65$). The objective measures evaluated in this work correlate better with the similarity evaluations. The best sample correlation with SIM ($|\bar{r}| = 0.86$) is obtained with the FWS measure calculated based on the internal STRAIGHT representation and the best sample correlation with NAT ($|\bar{r}| = 0.69$) with MCD calculated based on the synthesised output.

When noise-corrupted or enhanced samples or synthesis models adapted with noise-corrupted or enhanced data were used, the samples were evaluated in three scales (Table 1). The sample correlations between the objective measures and subjective evaluations are indicated with triangles in Figure 3. The notable quality difference between the natural and synthesised samples dominates the SIM and NAT evaluations which are exceptionally coherent across the test conditions ($|\bar{r}| = 0.97$). The best correlation with SIM ($|\bar{r}| = 0.90$), NAT ($|\bar{r}| = 0.89$), and BAK ($|\bar{r}| = 0.41$) is obtained with LLR.

While correlation between the objective evaluations and BAK appears weak when examined over the complete test set, background intrusiveness has a notable contribution to the objective scores. The sample correlation calculated for objective measures and BAK $|r(n)| \geq 0.93$ within the natural sample set and $|r(n)| \geq 0.70$ within the synthesised samples that represent an adapted model. This suggests that the objective measures emphasise speech qualities but are not invariant to background intrusiveness.

4. Discussion

4.1. Main results

We evaluated the correlation between several objective measures and subjective listening test results in two tasks. FWS calculated based on the internal spectral representation and LLR resulted in the best overall correlation with the subjective similarity scores. As discussed in [8], the correlation between the objective measures and subjective evaluations varies from voice to voice, but the sample correlation calculated between SIM and FWS int for individual voices in either dataset $|r(n)| > 0.75$ and the sample correlation between SIM and LLR measures $|r(n)| > 0.73$. FWS calculated based on the synthesised output also correlated well with SIM evaluations, and we note that FWS and LLR performed well also in the speech enhancement evaluation [6].

In the previous studies [9, 6, 10], the best correlation with subjective listening test results has been obtained with PESQ [5]. This is a standardised measure that incorporates perceptual and cognitive models for speech quality assessment. Despite the success obtained with PESQ, we believe the need for license-free evaluation measures remains. With projects like Simple4All [1], HMM-TTS is becoming more accessible for languages that are under-resourced both in terms of data an funding.

4.2. Similarity and naturalness

An ideal measure for objective evaluation would take into account all the factors that influence subjective listening test results, but this is a difficult task. For example, a smooth and natural voice is often rated better than a rougher voice that is otherwise more similar to the target speaker. Evaluation in a noisy background adds to the complexity as the increased background noise can mask synthesis artifacts and make a synthesised voice sound better.

The objective quality measures evaluated in this work operate on a one-dimensional scale whereas the human listeners rated the samples on based on two or three specific features. This is necessary when several factors affect the perceived overall quality. Hu and Loizou [6] used linear combinations of the basic objective measures to calculate composite measures tuned for the separate subjective scales. The measures evaluated in this work are, however, very correlated, which means nonlinear combinations should be used in order to introduce notable improvement compared to the best individual measures. The similarity aspect could also be assessed with speaker recognition techniques, for example.

4.3. Reference data

The objective measures evaluated in this work represent the so called intrusive or full-reference measures that require a target sample for comparison. Therefore the synthesised samples had to be generated with the alignments extracted from the target signal. Möller et al. [8] compared three non-intrusive measures in speech transmission and speech synthesis evaluation task, but concluded that the objective measures were not sufficiently accurate in predicting differences between synthesised speech quality.

Developing model-based measures for speaker similarity and naturalness would allow us to evaluate the quality of synthetic speech with less than perfect time alignment between the reference and synthetic stimulus. This will be growingly more desirable as synthetic speech tries to reproduce the prosodic aspects of the speech, including accent and speaking rhythm. Evaluating prosody with objective measures is not a realistic goal in the near future, but as the development prosody generation and spectral envelope synthesis can not be done completely separately, also the objective evaluation of the spectral envelope should be done in conjunction with model-generated prosody.

5. Conclusions and future work

We analysed correlation between objective measures and subjective listening test results in speaker adaptation task in the HMM-TTS framework. The measures were correlated with the subjective speaker similarity more than naturalness or background quality, and the best measures were FWS and LLR. While the measures studied in this work cannot replace subjective evaluation, the measures could be used to optimise the system parameters in a manner that better corresponds to listener preference. For example, speech enhancement parameters and regression tree size in adaptation are usually hand-tuned based on performance over some development data, and any measure can be used for the performance evaluation. MCD can also be used to optimise the adaptation transformations as proposed in [20].

Our datasets were quite small, and we would like to continue the verification process using larger datasets. We think objective evaluation should not require aligned samples, and therefore hope to investigate if test and reference samples could be compared based on local alignments, and if speaker recognition technologies could provide measures that correlate with the similarity scores in our listening tests.

[1] http://www.simple4all.org

6. Acknowledgements

This work received financial support from the Academy of Finland under the grants no 135003, 140969, and 251170, from Tekes under Perso and Funesomo projects, and from EC FP7 under grant agreement 287678. R. Karhila was supported by Langnet graduate school and Nokia Foundation. We acknowledge the computational resources provided by Aalto Science-IT project.

7. References

[1] H. Zen, K. Tokuda, and A. W. Black, "Statistical parametric speech synthesis," *Speech Communication*, vol. 51, no. 11, pp. 1039–1064, 2009.

[2] J. Yamagishi, B. Usabaev, S. King, O. Watts, J. Dines, J. Tian, Y. Guan, R. Hu, K. Oura, Y. J. Wu, K. Tokuda, R. Karhila, and M. Kurimo, "Thousands of voices for HMM-based speech synthesis-analysis and application of TTS systems built on various ASR corpora," *IEEE Trans. Audio, Speech, and Language Processing*, vol. 18, no. 5, pp. 984–1004, 2010.

[3] R. A. Clark, M. Podsiadlo, M. Fraser, C. Mayo, and S. King, "Statistical analysis of the Blizzard Challenge 2007 listening test results," in *Proc. Blizzard Challenge Workshop*, 2007.

[4] J. M. Tribolet, P. Noll, B. J. McDermott, and R. E. Crochiere, "A study of complexity and quality of speech waveform coders," in *Proc. ICASSP*, 1978, pp. 586–590.

[5] ITU-T, *Recommendation P.862 (02/2001) Perceptual evaluation of speech quality (PESQ): An objective method for end-to-end speech quality assessment of narrow-band telephone networks and speech codecs*.

[6] Y. Hu and P. Loizou, "Evaluation of objective quality measures for speech enhancement," *IEEE Trans. Audio, Speech, and Language Processing*, vol. 16, no. 1, pp. 229 –238, jan. 2008.

[7] N. Kitawaki and H. Nagabuchi, "Quality assessment of speech coding and speech synthesis systems," *IEEE Communications Magazine*, vol. 26, pp. 36–44, 1988.

[8] S. Möller, D. S. Kim, and L. Malfait, "Estimating the quality of synthesized and natural speech transmitted through telephone networks using single-ended prediction models," *Acta Acustica united with Acustica*, vol. 94, pp. 21–31, 2008.

[9] M. Cernak and M. Rusko, "An evaluation of synthetic speech using the pesq measure," in *Proc. European Congress on Acoustics*, 2005, pp. 2725–2728.

[10] D. Y. Huang, "Prediction of perceived sound quality of synthetic speech," in *Proc. APSIPA*, 2011.

[11] R. Karhila, U. Remes, and M. Kurimo, "HMM-based speech synthesis adaptation using noisy data: Analysis and evaluation methods," in *Proc. ICASSP*, 2013.

[12] R. Karhila, R. S. Doddipatla, M. Kurimo, and P. Smit, "Creating synthetic voices for children by adapting adult average voice using stacked transformations and VTLN," in *Proc. ICASSP*, 2012.

[13] J. Yamagishi, T. Kobayashi, Y. Nakano, K. Ogata, and J. Isogai, "Analysis of speaker adaptation algorithms for HMM-based speech synthesis and a constrained SMAPLR adaptation algorithm," *IEEE Trans. Audio, Speech, Lang. Process*, vol. 17, no. 1, pp. 66–83, 2009.

[14] J. Hansen and B. Pellom, "An effective quality evaluation protocol for speech enhancement algorithms," in *Fifth International Conference on Spoken Language Processing*, vol. 7, 1998, pp. 2819–2822.

[15] F. Itakura, "Minimum prediction residual principle applied to speech recognition," *IEEE Trans. Acoustics, Speech, and Signal Processing*, vol. 23, pp. 67–72, 1975.

[16] D. H. Klatt, "Prediction of perceived phonetic distance from critical-band spectra: a first step," in *Proc. ICASSP*, 1982, pp. 1278–1281.

[17] P. Loizou, "COLEA: A MATLAB tool for speech analysis," 1998.

[18] H. Kawahara, I. Masuda-Katsuse, and A. Cheveigné, "Restructuring speech representations using a pitch-adaptive time-frequency smoothing and an instantaneous-frequency-based F0 extraction: possible role of a repetitive structure in sounds," *Speech Communication*, vol. 27, pp. 187–207, 1999.

[19] M. Brookes, "VOICEBOX: Speech processing toolbox for MATLAB," 1998.

[20] L. Qin, Y. J. Wu, Z. H. Ling, R. H. Wang, and L. R. Dai, "Minimum generation error linear regression based model adaptation for HMM-based speech synthesis," in *Proc. ICASSP*, 2008, pp. 3953–3956.

Experiments with Signal-Driven Symbolic Prosody for Statistical Parametric Speech Synthesis

Fabio Tesser, Giacomo Sommavilla, Giulio Paci, Piero Cosi

Institute of Cognitive Sciences and Technologies, National Research Council, Padova, Italy
{fabio.tesser, giacomo.sommavilla, giulio.paci, piero.cosi}@pd.istc.cnr.it

Abstract

This paper presents a preliminary study on the use of symbolic prosody extracted from the speech signal to improve parameters prediction on HMM-based speech synthesis. The relationship between the prosodic labelling and the actual prosody of the training data is usually ignored in the building phase of corpus based TTS voices. In this work, different systems have been trained using prosodic labels predicted from speech and compared with the conventional system that predicts those labels solely from text. Experiments have been done using data from two speakers (one male and one female). Objective evaluation performed on a test set of the corpora shows that the proposed systems improve the prediction accuracy of phonemes duration and F0 trajectories. Advantages on the use of signal-driven symbolic prosody in place of the conventional text-driven symbolic prosody, and future works about the effective use of these information in the synthesis stage of a Text To Speech systems are also described.

Index Terms: statistical parametric speech synthesis, HMM-based speech synthesis, prosody prediction, symbolic prosody, ToBI.

1. Introduction

Modern TTS systems consist of two modules. The first one (called NLP or front end module) processes the input text and extracts a symbolic phonetic/linguistic representation of the utterance. The second one is the waveform generation module, that receives data from the front end and takes care of generating the audio signal. In statistical parametric synthesis systems, the waveform generation module incorporates acoustic models. They are trained using both linguistic information (evaluated by the NLP module) and parameters extracted from the speech signal.

Some of the most important linguistic features used by current statistical parametric speech synthesis systems are listed in [1]. The work presented here focuses on those features referring to the category of symbolic prosody, that is a compact representation useful to describe how the prosodic parameters vary inside an utterance. These features, called prosodic labels, have the peculiarity of representing speech properties belonging to both acoustic and symbolic linguistic domains.

For example, the ToBI standard [2] represents prosody using *break indices* that describe the degree of disjuncture between consecutive words and the tones associated with *phrase boundaries* and *pitch accents*.

While other features (like phonetic features, syllable features, part of speech, ...) depend on linguistic rules that apply solely to textual information, the symbolic prosody is also strongly related to the way in which the speaker has uttered the sentence. However, since the input of a TTS is text, usually symbolic prosody is evaluated only from text [3], using both handwritten rules or statistical methods.

Recently, researchers have investigated different methods for symbolic prosody extraction from the speech signal [4, 5, 6] in the field of speech analysis and recognition. The symbolic prosody evaluated from the actual speech signal, as opposed to the *text-driven* symbolic prosody, will be referred to as *signal-driven* symbolic prosody in this paper.

The purpose of this work is to investigate how the use of *signal-driven* prosodic information can improve the naturalness of parametric speech synthesis. This is determined experimentally by building different HMM-based systems that use different symbolic prosody estimation strategies, and comparing the parameter predictions with an objective assessment on a test set.

The work presented here is a preparatory study on the use of *signal-driven* symbolic prosody in TTS systems. The objective evaluation is important in this preliminary analysis stage because it is an indicator of what improvement on parameters prediction accuracy could be achieved if the *signal-driven* symbolic prosody was used in the synthesis phase of a Text To Speech system.

Anyway, this paper does not propose a technique ready for a TTS system, because the prediction of the prosodic labels from text is missing in this work.

However, this study is a first step towards the creation of a *signal-driven* symbolic prosody predictor from text, trained with linguistic features and *signal-driven* prosodic labels extracted respectively from text and audio data of a TTS speech corpus.

Therefore, the main advantage of the *signal-driven* symbolic prosody in TTS systems is that the prosodic labels are consistent with the speech corpus. Consequently it will be possible to model and predict the symbolic prosody of a specific speaker, or his particular speaking style used in the corpus.

The paper is organised as follows: Section 2 presents a discussion on how the paper's contributions are related to prior work in the field; Section 3 describes the tool used to extract the *signal-driven* symbolic prosody from the speech corpus; the different systems built, the experimental settings and the results are described on Section 4; finally, Section 5 concludes the paper and proposes some future developments.

2. Motivation

Efforts in the TTS field are always aiming at improving the naturalness of synthetic speech; one of the key challenges is the prediction of prosody and in particular on the fundamental frequency (F0). Research in this area is trying to improve the accuracy of estimates for this task, investigating new models for F0 [7, 8, 9], using different training methods [10], or experi-

menting on new topologies of the multi stream model used in classical HMM-based speech synthesis systems [11].

Other research works investigate on how different symbolic linguistic features can improve TTS quality; for example [12] reported on an investigation on how high level linguistic features extracted from text can improve the quality of prosody modelling, and [13] analysed the identification and generation (from text) of prosodic prominence in HMM-based TTS synthesis.

Symbolic prosody is a default feature used in standard training of a HMM-Based system [1]. The prosodic labels are assigned according to information extracted from text [3].

Regarding the use of symbolic prosody in the training phase of statistical parametric speech synthesis, the assumption is that there is some consistent relationship between the prosodic labelling and the actual prosody of the training data. This assumption is not always true if the symbolic prosody is predicted only from text. In fact, because the symbolic prosody is also linked to acoustical parameters, it is possible for different prosodic labels to be associated to the same sentence, uttered by different speakers (between-speakers variability). Moreover, prosodic labels may also change depending on how a single speaker pronounces the same sentence (within-speaker variability).

In all the works cited above the features used are totally extracted from text, ignoring relations with the acoustic signal of the corpus used for the training. In fact, classic corpus-based voice building methods do not care if the sentences of the training corpus are actually uttered by the speaker according to the particular prosody described by text-predicted prosodic labels.

On the contrary, the procedure presented in this paper extracts the symbolic prosody features from the speech signal, in order to make use of the relationship between the prosodic labelling and the actual prosody of the training data.

This relationship has been investigated in [14], where an HMM-based TTS system built with hand annotated labels of ToBI events obtained the best result on the evaluation test.

However, differently from that experiment, this paper proposes the use of tools that automatically extract the symbolic prosody directly from audio, making the procedure reproducible in several TTS corpora, without the need of manual annotation. The hypothesis of this work is that the HMM models can improve TTS quality if trained with *signal-driven* prosodic labels that are supposedly more coherent with the audio samples of the corpus than the text-predicted labels.

This assumption is similar to the one that motivates the use of multiple pronunciation words in phonetized lexicon. In that case, a speaker could have uttered a word with phonemes that are different from those expected by the TTS training system. Ignoring this difference leads to a bad training of the models. In this case, being able to automatically recognize which phoneme has been actually pronounced by the speaker allows to build a system which can train HMM models with more appropriate phonetic labels. Similarly, the work presented here studies the possibility to train the models of a TTS system with the prosodic labels that best describe the actual statement of the speaker.

3. Signal-driven symbolic prosody

In order to compute the *signal-driven* symbolic prosody, it has been decided to use the AuToBI system [5], because it is a publicly available tool for automatic detection and classification of the prosodic events that adheres to the ToBI annotation standard used in many TTS front-end.

AuToBI operates by predicting prosodic events at the word level using the speech signal and its word segmentation. The generation of the hypothesized ToBI labels consists of different tasks of tones' detection and classification using models trained on prosodic annotated corpora.

The accuracy of the detection and classification tasks has been evaluated in [5], reporting good results for pitch accent and satisfactory results for phrase boundaries.

Using the *signal-driven* symbolic prosody, instead of text driven prosody, within the training stage of a corpus based TTS system, the actual prosody of the training data is taken into consideration.

Table 1 shows a sentence and two ToBI transcriptions: the first one is predicted using a linguistic front-end that makes use only of the text, while the second one was obtained using AuToBI and the speech signal. The first transcription depends only on the text, regardless of the particular pronunciation. On the other hand, the second one can highlight peculiar prosodic events actually uttered by the speaker.

Text	"Tom	Spink	has	a	harpoon."
Transcription 1	L+H*	L+H*			!H* L-L%
Transcription 2	L* H-	L* L-			L* L-L%

Table 1: *Two ToBI transcriptions of the same sentence, the former is predicted from text, the latter from speech signal using AuToBI.*

4. Experiments

4.1. Systems Built

All systems have been built using a modified version of MaryTTS 5.0 [15] as linguistic front end for extracting monophone and full context labels, while the phonetic alignment has been done using HTK 3.4.1 [16].

The HTS HMM speech synthesis toolkit version 2.2 [17] has been used for building the models; mgc (mel-generalised cepstrum) spectral parameters and voicing strengths for mixed excitation [18] are modelled using continuous probability distribution, while logF0 parts are modelled using the multi-space probability distribution (MSD) [19]. The systems have been built using the default speaker-dependent parameters of HTS: i) decision tree based state clustering; ii) separate streams to model each of the static, delta and delta-delta features; iii) single Gaussian models.

The following three systems have been built for the evaluation.

4.1.1. BASE system

The baseline system uses the *text-driven* symbolic prosody computed by the MaryTTS linguistic front-end. This component contains handwritten rules that uses punctuation marks and word's POS information to determine the prosodic labels.

4.1.2. FULL system

The FULL system uses all the prosodic labels computed by AuToBI, including pitch accent, boundary tones and implicitly also the break index associated with tones. This means that the phrase splitting is controlled by AuToBI and not by punctuation.

4.1.3. P-ACC system

This system is an hybrid system between BASE and FULL; in this case the boundary tones and the phrasing is controlled by the MaryTTS linguistic front-end, while the pitch accents are assigned by AuToBI. The P-ACC system has been created because AuToBI has proven to give slightly worse prediction results of phrase boundaries than pitch accents [5].

4.2. Experimental settings

The systems described above have been evaluated on two CMU ARCTIC speech synthesis data sets [20]. A U.S. female English speaker (slt) and a U.S male English speaker (bdl) were used. Each data set contains recordings of the same 1132 phonetically balanced sentences, totalling about 0.95 hours of speech per speaker. To obtain objective accuracy measures, 300 sentence has been used only for the test and the remaining 832 were used as training set.

Audio has been sampled at 16 kHz, the speech features used were mel generalized cepstral coefficients of order 24 and band-pass voicing strengths of order 5.

The AuToBI models used have been trained on U.S. English speech. The z-score normalization approach has been used for normalizing pitch and intensity values across speakers.

4.3. Evaluation indicators

The accuracy of the proposed technique has been evaluated objectively for each parameter x, comparing the predicted parameter x_P and the natural one x_N.

The generation of the parameters has been done using the maximum likelihood parameter generation algorithm where the state sequence is given (case 1 of [21]), including global variance [22].

The comparison has taken into consideration the following indicators: i) the *root mean square error* (RMSE) as measure of the prediction error on the parameters; ii) the *correlation coefficient* (ρ) as measure of the similarity between the two parameter trajectories; iii) the average number of leaf nodes (LEAVES) on the clustered trees which represents model complexity [11].

In the case of F0, mgcep and strengths (all except the duration), the parameters taken into consideration are time trajectories. In these cases, model-level alignments given by label files of natural speech have been applied in order to easily compare the generated trajectories between natural speech and generated speech.

4.4. Results

4.4.1. Duration

The model of duration is evaluated at the phoneme level, in this case x_p is the duration of the predicted phoneme and x_N is the duration of the natural one. Table 2 shows the result of the objective comparisons among the three systems. It can be observed that the P-ACC system is the best for both speakers. The number of leaves of the clustered trees shows an increase of the model complexity for the FULL system with respect to the BASE system.

4.4.2. F0

Because F0 is continuous in voiced regions and undefined in unvoiced regions, also the *voicing classification error* (VCE) is taken into accout. VCE, like in [7], is defined as the rate of mismatched voicing label, and can be written as:

SYS	RMSE (ms)		ρ		LEAVES	
	bdl	slt	bdl	slt	bdl	slt
BASE	30.7	33.8	0.714	0.793	440	436
P-ACC	29.7	33.5	0.740	0.799	451	425
FULL	30.0	33.8	0.733	0.794	455	447

Table 2: *Evaluation indices and model complexity for the phoneme duration prediction, evaluated in the test set.*

$$VCE = 100 \frac{\sum_{t=t_1}^{t_N} \delta\left(v_{x_P}(t) \neq v_{x_N}(t)\right)}{N}, \quad (1)$$

where v_{x_P} is the voiced binary index, that is equal to 1 if the frame is voiced, 0 otherwise.

Table 3 shows the result of the objective comparisons among the three systems for the task of F0 prediction. *Signal-driven* symbolic prosody (P-ACC and FULL) systems show an improved accuracy on prosody prediction for what concerns RMSE and correlation coefficients with respect to the BASE system. A weak preference for P-ACC system against FULL system can be given according to these two indices. The VCE index shows a preference for FULL for bdl speaker and no significant preference for slt speaker. Also in this case the LEAVES indicator shows that the FULL models are more complex than those of system BASE, while there are not significant differences between P-ACC and BASE.

SYS	RMSE (Hz)		ρ	
	bdl	slt	bdl	slt
BASE	19.4	13.8	0.628	0.738
P-ACC	18.9	13.6	0.642	0.748
FULL	19.1	13.7	0.632	0.744
SYS	VCE (%)		LEAVES	
	bdl	slt	bdl	slt
BASE	8.4	7.1	457	439
P-ACC	8.6	7.1	456	436
FULL	8.3	7.2	467	464

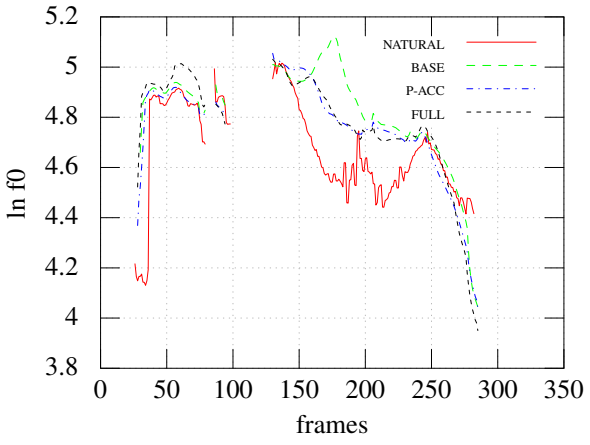
Table 3: *Evaluation indices and model complexity for F0 prediction, evaluated in the test set.*

A visual example of the generated pitch contours is illustrated in Figure 1, where it is plotted a comparison among F0 generation by system BASE, P-ACC and system FULL and the natural speech. Both figures, the first for the male speaker and the second for the female one, show that systems P-ACC and FULL predict the pitch accent F0 values in the middle of the sentence (frames 150-200 for bdl, frames 200-250 for slt) more accurately compared to the baseline.

4.4.3. Mgc and strength coefficients

In the cases of mgc and strength coefficients, the features taken into consideration are multidimensional (size 24 for mel-generalised cepstral coefficients and size 5 for strength coefficients), so the z-score normalization of each coefficient i has been computed:

$$z_i = \frac{x_i - \mu_i}{\sigma_i}; \quad (2)$$



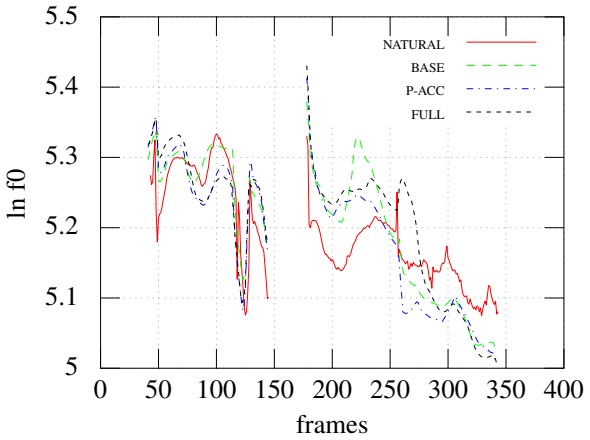


Figure 1: *Two examples showing F0 trajectories generated by system BASE, system P-ACC and system FULL compared to the pitch extracted from the audio (NATURAL). The two plots refers to samples taken from bdl (male) and slt (female) test sets, respectively.*

the RMSE of each coefficient has been evaluated using this normalized scale, in order to average these values and to present an unique value (RMSE (z)).

Tables 4 and 5 show the results of the objective comparisons among the three systems for mel-generalised cepstral coefficients and strength coefficients. It can be observed that, with the exception of strength coefficient for the speaker bdl (where the system FULL improved the accuracy), no significant differences for these indicators can be appreciated with respect to the system BASE.

4.5. Discussion

As seen in the above results, *signal-driven* symbolic prosody systems (P-ACC, FULL) improve the accuracy on both duration and pitch prediction with respect to text-driven symbolic prosody system (BASE). On these tasks P-ACC performs better than FULL, possibly because the task of automatic boundary tones detection and classification is more difficult than that of pitch accent detection and classification. Actually, also AuToBI

SYS	RMSE (z)		ρ		LEAVES	
	bdl	slt	bdl	slt	bdl	slt
BASE	0.81	0.73	0.646	0.719	196	200
P-ACC	0.81	0.73	0.647	0.720	195	200
FULL	0.81	0.73	0.647	0.719	194	197

Table 4: *Evaluation indices and model complexity for mel-generalised cepstral coefficients prediction, evaluated in the test set.*

SYS	RMSE (z)		ρ		LEAVES	
	bdl	slt	bdl	slt	bdl	slt
BASE	0.87	0.82	0.622	0.634	91	84
P-ACC	0.89	0.82	0.623	0.635	89	85
FULL	0.85	0.82	0.622	0.635	91	84

Table 5: *Evaluation indices and model complexity for strength coefficients prediction, evaluated in the test set.*

shows slightly worse results in the task of automatic boundary tones detection [5].

Results on spectral features show no significant difference between the proposed systems and the baseline.

With respect to the baseline system, the *signal-driven* symbolic prosody systems bring an improvement that is more evident for the male speaker than for the female one.

This could depend on the fact that the AuToBI models used for the symbolic prosody prediction have been trained on more male than female speakers or they have been trained on speakers more prosodically similar to bdl than slt.

5. Conclusions & further work

This paper has compared the use of *signal-driven* symbolic prosody to the classical method (that extracts the symbolic prosody from text) on the training stage of statistical parametric speech synthesis. Two *signal-driven* symbolic prosody systems have been built using labels computed from the AuToBI system; these systems have been compared to the classical one. Objective measures have shown that the use of *signal-driven* symbolic prosody during the training of HMM-based TTS system improves the prediction of duration and pitch trajectories.

The effective utilisation of symbolic prosody within a TTS system, however, requires to predict the symbolic prosody from text.

Future investigations in this direction will be aimed to create a statistically based predictor of symbolic prosody from text but tuned on the specific acoustic parameters of the TTS corpus. Such predictor will be trained on *signal-driven* prosodic labels extracted from the speech corpus with AuToBI (or with a different *signal-driven* prosody tool), and it can be implemented as a classifier that uses as input the linguistic features extracted from text. If the accuracy of the classifier will be precise enough then it will be able to better represent the prosody in the corpus with respect to the classical predictor that only uses text. Subsequently the improvement described in the this work can be applied to every text input, and the benefit of a speaker-dependent symbolic prosody classifier (i.e. built with data from a single speaker) will be to make it possible for the statistical models to better fit the prosodic style of that particular speaker.

6. Acknowledgements

Thanks to Andrew Rosenberg, Marc Schröder, the MaryTTS team and the HTS team. This research was partly funded by EU-FP7 project ALIZ-E (ICT-248116).

7. References

[1] H. Zen, K. Tokuda, and A. W. Black, "Statistical parametric speech synthesis," *Speech Communication*, vol. 51, no. 11, pp. 1039–1064, Nov. 2009.

[2] K. Silverman, M. Beckman, J. Pitrelli, M. Ostendorf, C. Wightman, P. Price, J. Pierrehumbert, and J. Hirschberg, "ToBI: A standard for labeling English prosody," in *Second International Conference on Spoken Language Processing*, vol. 2, no. October, 1992, pp. 867–870.

[3] K. Ross and M. Ostendorf, "Prediction of abstract prosodic labels for speech synthesis," *Computer Speech & Language*, vol. 10, no. 3, pp. 155–185, Jul. 1996.

[4] K. Chen, M. Hasegawa-Johnson, and A. Cohen, "An automatic prosody labeling system using ANN-based syntactic-prosodic model and GMM-based acoustic-prosodic model," in *ICASSP*, 2004, pp. 509–512.

[5] A. Rosenberg, "AuToBI A Tool for Automatic ToBI annotation," in *Interspeech*, September 2010, pp. 146–149.

[6] J. H. Jeon and Y. Liu, "Automatic prosodic event detection using a novel labeling and selection method in co-training," *Speech Communication*, vol. 54, no. 3, pp. 445–458, Mar. 2012.

[7] K. Yu and S. Young, "Continuous F0 Modeling for HMM Based Statistical Parametric Speech Synthesis," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 5, pp. 1071–1079, Jul. 2011.

[8] K. Yu and S. Young, "Joint modelling of voicing label and continuous F0 for HMM based speech synthesis," in *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, May 2011, pp. 4572–4575.

[9] T. Koriyama, T. Nose, and T. Kobayashi, "Discontinuous Observation HMM for Prosodic-Event-Based F0 Generation," in *Interspeech*, 2012.

[10] J. Latorre, M. Gales, and H. Zen, "Training a parametric-based logF0 model with the minimum generation error criterion," in *Proceedings of the Interspeech*, September 2010, pp. 2174–2177.

[11] T. Koriyama, T. Nose, and T. Kobayashi, "An F0 modeling technique based on prosodic events for spontaneous speech synthesis," in *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, Mar. 2012, pp. 4589–4592.

[12] N. Obin, P. Lanchantin, M. Avanzi, A. Lacheret-dujour, and X. Rodet, "Toward Improved HMM-based Speech Synthesis using High-Level Syntactical Features," in *Speech Prosody 2010 Proceeding*, 2010.

[13] L. Badino, R. Clark, and M. Wester, "Towards Hierarchical Prosodic Prominence Generation in TTS Synthesis," in *INTERSPEECH*, 2012.

[14] O. Watts, J. Yamagishi, and S. King, "The role of higher-level linguistic features in HMM-based speech synthesis," in *Interspeech*, September 2010, pp. 841–844.

[15] M. Schröder, M. Charfuelan, S. Pammi, and I. Steiner, "Open source voice creation toolkit for the MARY TTS Platform," in *Interspeech*, no. ii, 2011.

[16] S. Young, G. Evermann, D. Kershaw, G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev, and P. Woodland, "The HTK book (for HTK version 3.2)," Cambridge University, Tech. Rep. July 2000, 2002.

[17] H. Zen, T. Nose, J. Yamagishi, S. Sako, T. Masuko, A. Black, and K. Tokuda, "The HMM-based speech synthesis system (HTS) version 2.0," in *The 6th International Workshop on Speech Synthesis*. Citeseer, 2007, pp. 294–299.

[18] T. Yoshimura, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, "Mixed Excitation for HMM-based Speech Synthesis," in *Eurospeech*, 2001.

[19] K. Tokuda, T. Masuko, N. Miyazaki, and T. Kobayashi, "Hidden Markov models based on multi-space probability distribution for pitch pattern modeling," in *1999 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings. ICASSP99 (Cat. No.99CH36258)*. IEEE, 1999, pp. 229–232 vol.1.

[20] J. Kominek, A. Black, and V. Ver, "CMU ARCTIC databases for speech synthesis," CMU, Tech. Rep., 2003.

[21] K. Tokuda, T. Yoshimura, T. Masuko, T. Kobayashi, and T. Kitamura, "Speech parameter generation algorithms for HMM-based speech synthesis," in *2000 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings (Cat. No.00CH37100)*, vol. 3. IEEE, 2000, pp. 1315–1318.

[22] T. Toda and K. Tokuda, "A speech parameter generation algorithm considering global variance for HMM-based speech synthesis," in *IEICE*, no. 5, 2007, pp. 816–824.

Significance of word-terminal syllables for prediction of phrase breaks in Text-to-Speech systems for Indian languages

Anandaswarup Vadapalli[1], *Peri Bhaskararao*[1], *Kishore Prahallad*[1]

[1] Speech and Vision Lab, IIIT Hyderabad, India

anandaswarup.vadapalli@research.iiit.ac.in, bha.peri@iiit.ac.in, kishore@iiit.ac.in

Abstract

Phrase break prediction is very important for speech synthesis. Traditional methods of phrase break prediction have used linguistic resources like part-of-speech (POS) sequence information for modeling these breaks. In the context of Indian languages, we propose to look at syllable level features and explore the use of word-terminal syllables to model phrase breaks. We hypothesize that these terminal syllables serve to discriminate words based on syntactic meaning, and can therefore be used to model phrase breaks. We utilize these terminal syllables in building models for automatic phrase break prediction from text and demonstrate by means of objective and subjective measures that these models perform as well as traditional models using POS sequence information. Thus the proposed method avoids the need for POS taggers for prosodic phrasing in Indian languages.

Index Terms: Phrase Breaks, Word-Terminal Syllables, Text-to-Speech

1. Introduction

Phrase break prediction plays an important role in the context of speech synthesis. It is known that phrase breaks have a non-linear relationship with syntactic breaks [1]. It is also known that phrase breaks are specific to a speaker [2] [3].

Phrase breaks are manifested in the speech signal in the form of several acoustic cues like pauses as well as relative changes in the intonation and duration of syllables. Acoustic cues such as pre pausal lengthening of rhyme, speaking rate, breaths, boundary tones and glottalization also play a role in indicating phrase breaks in speech [4], [5], [6]. However, representing these non pause acoustic cues in terms of features is not easy and not well understood [2]. In this paper we restrict ourselves only to pauses in speech, and limit our phrase break models to predicting the locations of pauses while synthesizing speech. This is the approach followed in [7] and [8].

Traditional methods of phrase break prediction have used linguistic resources like part-of-speech (POS) sequences generated by POS taggers or shallow parsers to model phrase breaks. Many different machine learning algorithms have been applied to phrase break prediction; for example, decision trees [9], [10]; n-gram models [1], [11]; finite state transducers [12] and memory based learning [13]. However, regardless of the machine learning technique used, the primary feature used by the classifier has been POS tags.

In all the approaches mentioned above, POS tags are directly used as input into the phrase break classifier. In Parlikar and Black [7] they are used to construct grammar based parse trees, which in turn provides features for a decision tree based phrase break predictor. Previous work therefore suggests that POS tagging is a necessary first step in phrase break prediction.

All these traditional methods assume the availability of hand labeled training data, or high quality POS taggers/shallow parsers which can generate POS tags for the training data with a high level of accuracy. As a result, these methods can not be used for languages where the necessary linguistic resources are not readily available, and manual annotation of data is expensive and time consuming.

In view of the above limitations, there has recently been a lot of interest in unsupervised methods of inducing word representations which can be used as surrogates for POS tags, in the phrase break prediction task. Parlikar and Black [8] used the Ney-Essen clustering algorithm [14] to automatically induce POS tags. These induced POS tags are automatically generated from text using the frequency analysis of the words. However this approach faces an issue when applied to Indian languages, which are agglutinative in nature. In these languages words are formed by joining morphemes together. Moreover due to the postpositional nature of these languages, syllable level suffixes get attached to the ending of words. These suffixes give specific syntactic meaning in terms of tense, gender etc. These characteristics of Indian languages result in an increase in vocabulary size, i. e. the number of words. Thus it is hard to work with scripts of Indian languages using a word level representation.

A better solution may lie in dealing with sub-word units like syllables or multi-syllable units [15], [16], [17]. In [15], a set of *morpheme tags* units were manually identified and used to model phrase breaks. The *morpheme tags* consist of one or two syllables, typically found at the end of the word. The experiments were conducted on Telugu. Manual identification of this set of morpheme tags is hard and may require sufficient linguistic knowledge.

In the current work, we propose to look at syllable level units, and explore the terminal syllables of a word to model phrase breaks. A terminal syllable is the last syllable in the word. We hypothesize that these terminal syllables serve to discriminate words based on syntactic meaning, and that these terminal syllables can be used to model phrase breaks. We automatically identify the set of terminal syllables which could be used to model phrase breaks. We experiment our approach with six Indian languages, and report results on the automatic prediction of phrase breaks from the text using these terminal syllables. Finally, we incorporate the proposed phrase break model in a Text-to-Speech system, and demonstrate its usefulness with listening tests.

2. Database used in this study

In our study we look at two different corpora that have speech in different styles.

The ***IIIT-MCIT (Lenina)*** corpus is a corpus developed at IIIT Hyderabad, which was used to build a synthetic voice in Telugu. The corpus consists of text prompts taken from a set of popular children's stories in Telugu, and the corresponding recordings recorded in a story telling style in a clean studio environment. The corpus has 4043 text prompts and the audio size is about 6 hours. The style of the corpus is "story telling".

The ***IIIT-H Indic*** [18] database consists of text and speech data in Telugu, Hindi, Kannada, Tamil, Malayalam, Marathi and Bengali. Each language in the database consists of a set of 1000 text prompts selected from Wikipedia articles in the corresponding language, selected in such a way as to cover the 5000 most frequent words in the corresponding languages. The corresponding recordings were recorded by a native speaker of each language in a clean studio environment. On an average the size of the audio is about 1.5 hours for each language. The style of this corpus is "isolated sentences".

2.1. Annotation of phrase breaks

As we do not have a corpus with hand annotated phrase breaks, we derive the location and duration of the phrase breaks from the speech data. In order to derive the locations and durations of pauses introduced by the speaker, we force align the speech with the corresponding text prompts using the HMM tool in Festvox [19]. This gives us the locations of pauses introduced by the speaker while recording the utterances.

A question is what should be the duration of a pause to consider it as a phrase break? To answer the above question, we analyzed the durations of the pauses introduced by the speaker for both the *Lenina* and *IIITH-Indic* databases. Figure 1 shows the histogram plots of silence durations for all the six languages.

An analysis of the histogram plot shows that for Hindi the majority of the pauses are less than 80 ms in duration, for Telugu (IIITH-Indic) the majority of the pauses range from a few milliseconds to 480 ms, for Kannada and Tamil most of the pause durations range from a few milliseconds to 480 ms and for Bengali the pause durations range from a few milliseconds to 640 ms. In the case of the *Lenina* database, the pause durations range mainly from 80 ms to 640 ms.

We thus observe that the pause durations vary over a significant range within a language and also between languages. We experiment with different thresholds above which a pause is marked as a phrase break. We experiment with thresholds of 25 ms, 50 ms and 80 ms, whereby we mark all pauses with durations greater than the threshold as phrase breaks. We also experiment with the case where we mark all pauses as phrase breaks regardless of their duration.

3. Phrase Breaks vs. Syntactic Breaks

A question that is often asked is whether there is any relation between phrase breaks and syntactic breaks. While it is known that there is some correspondence between syntax and prosody, the relationship between them is not formally defined [1], [3]. We illustrate this by means of two examples.

Consider the Telugu sentence (represented in ITRANS transliteration scheme) shown in Table 1, taken from the *Lenina Database*, which has been annotated with the location of prosodic and syntactic breaks.

In a similar fashion consider the Hindi sentence shown in Table 2 taken from the *IIITH-Indic* database, which has also been annotated with the location of phrase and syntactic breaks.

From the above examples it is clear that while there is some correspondence between the phrase and syntactic breaks of an utterance, the relationship between them is not linear.

3.1. Syntactic breaks used in the study

Syntactic breaks were derived from text using the shallow parser [20] developed at IIIT Hyderabad. This tool uses conditional random fields (CRF) and transformational based learning (TBL) to perform chunking and POS tagging of text. In [20] the authors report accuracies of 77.37%, 78.66% and 76.08% for the chunking task and 79.15%, 80.97% and 83.74% for the POS tagging task, for the three languages Telugu, Hindi and Bengali respectively.

The location of syntactic breaks was derived by running the shallow parser on the text data. The tool parsed the text into syntactic constituents, and the end of each constituent was taken as a syntactic break.

3.2. Correlation between Phrase breaks and Syntactic breaks

In order to calculate the correlation between phrase and syntactic breaks, we conducted the following experiment for all languages under consideration: Telugu, Hindi, Kannada, Tamil and Bengali. For every word in each language, a binary feature which indicates the presence or absence of a break after that word, was derived. The presence of a break was indicated by 1 and the absence of a break by -1.

Let $\mathbf{S} = [s_1, ..., s_w, ..., s_N]$ denote the sequence of binary features derived for the words in the database using syntactic break information, where N denotes the total number of words in the database.

Let $\mathbf{P} = [p_1, ..., p_w, ..., p_N]$ denote the sequence of binary features derived using phrase breaks (pauses in speech), where N denotes the total number of words in the database. These breaks were derived from phrase break annotation described in 2.1.

The correlation coefficient between **S** and **P** is calculated using the following equation.

$$c(\mathbf{S}, \mathbf{P}) = \frac{\sum_{w=1}^{N}(s_w - \bar{s})(p_w - \bar{p})}{\sqrt{\sum_{w=1}^{N}(s_w - \bar{s})^2}\sqrt{\sum_{w=1}^{N}(p_w - \bar{p})^2}}$$

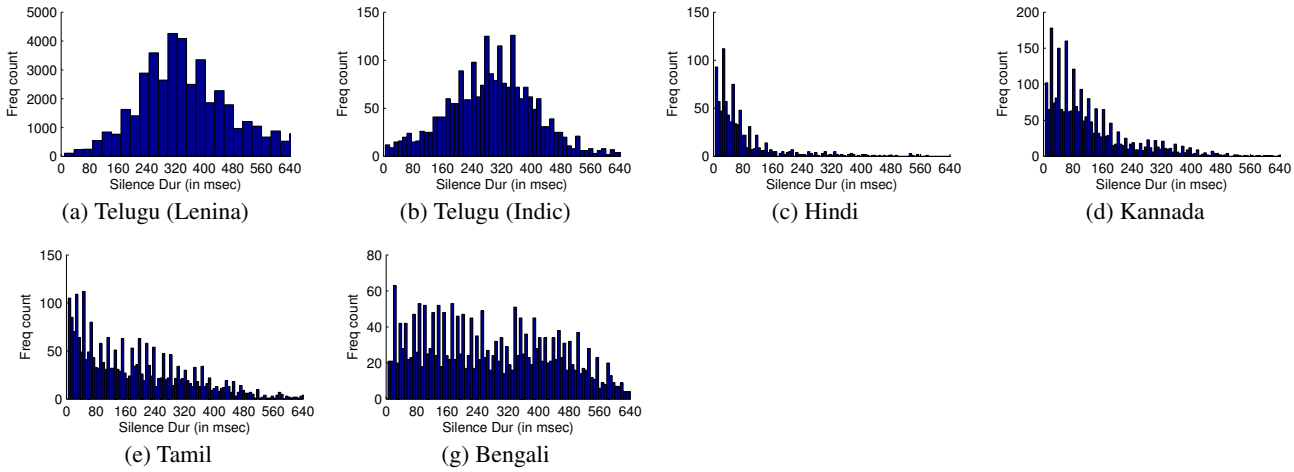
where $\bar{s}$ and $\bar{p}$ denote the mean values of **S** and **P** respectively.

Table 3 shows the correlation coefficients between syntactic and phrase breaks for the six languages.

Language	Correlation Coefficient
Telugu (Lenina)	0.26
Telugu (Indic)	0.27
Hindi	0.12
Kannada	0.29
Tamil	0.18
Bengali	0.20

Table 3: Correlation coefficients between syntactic breaks and phrase breaks for the six languages

An observation of the values in Table 3 shows that the values of correlation coefficients between syntactic breaks and phrase breaks, for all the languages, does not exceed 0.3. This indicates that there is a significant variation between syntactic and phrase breaks, in all the languages under consideration.



(a) Telugu (Lenina) (b) Telugu (Indic) (c) Hindi (d) Kannada

(e) Tamil (g) Bengali

Figure 1: Histograms of silence durations in all 6 languages.

"brahmadattud:u #B(260ms) kaashiiraajyaanni paripaalin:chei kaalan:loo #B(440ms) aa nagaran:loo #B(410ms) dhanikud:aina #B(360ms) oka goppavartakud:un:d:eivaad:u"
"brahmadattud:u | kaashiiraajyaanni | paripaalin:chei | kaalan:loo | aa nagaran:loo | dhanikud:aina | oka goppavartakud:un:d:eivaad:u"

Table 1: An example sentence in Telugu annotated with locations of phrase and syntactic breaks (the word-terminal syllables have been underlined) where *#B* denotes a phrase break and the numerical value in brackets denotes the break duration in milliseconds and | denotes a syntactic break

"san:bhava hai ki #B(80ms) isakaa #B(65ms) aavishhkaara #B(10ms) isasei bhii bahuta pahalei huaa hoo"
"san:bhava | hai | ki | isakaa aavishhkaara | isasei bhii | bahuta pahalei | huaa hoo"

Table 2: An example sentence in Hindi annotated with locations of phrase and syntactic breaks where *#B* denotes a phrase break and the numerical value in brackets denotes the break duration in milliseconds and | denotes a syntactic break

4. Correlation between word-terminal syllables and breaks

In order to model phrase breaks, we look at syllable level features. An examination of the Telugu example from Section 3 shows that a phrase break has occurred after two words ending in the syllable *loo*, and after a word ending in the syllable *d:u*. The utterance ending break has also occurred after a word ending in the syllable *d:u*. This motivated us to look at word-terminal syllables as a feature set which can be used to model phrase breaks. We performed an analysis, whereby the correlation between the word-terminal syllables and phrase breaks was studied. As part of this analysis we computed the conditional probability $p(\text{break} \mid \text{terminal syllable})$ as follows

$$p(\text{break} \mid \text{terminal syllable}) = \frac{N(\text{break, terminal syllable})}{N(\text{terminal syllable})},$$
$$\forall N(\text{terminal syllable}) > 50$$

Our analysis, showed that for a few terminal syllables, the probability of a word ending in that terminal syllable, preceding a phrase break, is high. That is, for a few terminal syllables the value of the conditional probability $p(\text{phrase break} \mid \text{terminal-syllable})$ is high. The value of this conditional probability tapers off beyond these top few terminal syllables. Table 4 shows the top terminal syllables, derived from this analysis for each of the six languages. The values in the parentheses are the values of $p(\text{phrase break} \mid \text{terminal-syllable})$ for those particular syllables.

As can be observed from the Table 4 we can see that the value of $p(\text{phrase break} \mid \text{terminal-syllable})$ for the top syllables are in the range 0.6 - 1.0. As a result, we hypothesize that these word terminal syllables are good candidates for a feature set to predict phrase breaks from text.

5. Prediction of phrase breaks from text

Prediction of prosodic phrase breaks from text can be achieved by building a phrasing model. Typically as a first approach a punctuation based phrasing model is used. The output of this model is then refined by using models built using POS tags and other linguistic information. However, text in Indian languages very rarely has any punctuations (except for sentence endings). Hence when dealing with Indian languages a simple punctuation based phrasing model will not work and more sophisticated phrasing models are required. These model can either be a set of heuristic rules or a machine learning model trained on features extracted from the text. Generally, the first step in building such phrasing models involves annotating the text with phrase breaks, which has been described in Section 2.1. This annotated text can be used in several ways. The text can be used to derive a set of heuristic rules, which can be used to derive the location of phrase breaks in the text. We can also extract several features from this text to train a machine learning model, which can be

Telugu(Lenina)	nai(0.98), buu(0.97), jaa(0.92), chchu(0.90), du(0.87), vaa(0.87), yyaa(0.85), daa(0.84), stei(0.83), yi(0.80), tei(0.80), di(0.79), chchi(0.79), t:ei(0.79), mmaa(0.78), d:u(0.78), llaa(0.75), chii(0.74), ppi(0.73), chi(0.73), ...
Telugu (Indic)	d:i(0.84), mu(0.82), loo(0.78), yi(0.77), du(0.76), di(0.70), nu(0.67), san:(0.65), llaa(0.64), d:u(0.63), vu(0.61), ru(0.58), d:aa(0.57), ki(0.54), ni(0.54), gaa(0.53), lu(0.53), ku(0.51), ran:(0.50), chi(0.48), ...
Hindi	hai(0.75), hain:(0.74), thaa(0.64), sha(0.43), da(0.40), ei(0.34), koo(0.26), yaa(0.24), nd~a(0.24), la(0.23), pa(0.23), kaa(0.22), ga(0.22), va(0.21), sa(0.21), na(0.20), ra(0.19), ti(0.18), kha(0.17), bhii(0.17), ...
Kannada	de(0.97), ki(0.93), lli(0.69), ru(0.67), re(0.65), nnu(0.62), l:u(0.61), gi(0.57), ge(0.57), da(0.54), ya(0.54), du(0.52), na(0.51), tra(0.50), ti(0.43), ga(0.43), vu(0.42), ttu(0.42), ka(0.41), ra(0.40), ...
Tamil	kum(1.00), n~ar(0.99), llai(0.96), lam(0.94), r:r:i(0.77), chan~(0.73), ng~kal:(0.72), thu(0.69), rai(0.67), than~(0.61), kal:(0.57), rkal:(0.54), ntha(0.53), n~r:u(0.48), yil(0.47), ththil(0.46), ththu(0.35), ya(0.35), ka(0.32), kku(0.32), ...
Bengali	hay(0.91), chhi(0.81), nya(0.78), sa(0.77), chhe(0.70), da(0.66), ja(0.60), naa(0.59), sha(0.58), ban:(0.57), i(0.57), ba(0.56), be(0.53), ke(0.51), na(0.51), t:a(0.50), re(0.50), nd~a(0.48), le(0.48), ga(0.48), ...

Table 4: Top word-terminal syllables for all the languages. The figure in brackets is $p(\text{phrase break} \mid \text{terminal syllable})$

used to predict phrase breaks from text.

We experiment with three different approaches and report the results of phrase break prediction from text, for both *Lenina* and *IIITH-Indic* databases. In our first approach we derive a simple rule which utilizes the syntactic break location (obtained from the shallow parser (Section 3)) along with terminal syllable information to derive the location of the phrase breaks in text. Our second approach utilizes terminal syllable information, which we extract from the text, to build a machine learning model for phrase break prediction. In our third approach, we use POS tag sequence information, (which we obtain from running the shallow parser over the text (Section 3)) to build a machine learning model for phrase break prediction.

As the text in these databases has already been annotated with prosodic phrase breaks, a ground truth to compute the performance of our approaches is available. We report the performance of our approaches in terms of the F-measure [21] which is defined as the harmonic mean of the precision and recall. F-measure values range from 0 to 1, with higher values indicating better performance.

5.1. Simple rule based phrase break prediction

We derive a simple rule to give us the location of phrase breaks in text. This rule uses syntactic break locations along with the terminal syllable information to derive the locations of the phrase breaks in text.

From our analysis of the correlation between terminal syllables and phrase breaks in Section 4 we observed that the top 50 word terminal syllables in each language have a high correlation of occurrence along with phrase breaks. We combined our knowledge of the syntactic break locations (derived from the shallow parser (Section 3)) with this observation to develop the following heuristic rule for phrase break prediction from text.

"If the word ending of a word in the text has been marked as a syntactic break and the last syllable of the word (the terminal syllable) is among the list of the top 50 terminal syllables for that language (derived from our analysis), then that syntactic break is also a phrase break."

We use this rule to derive the locations of the phrase breaks in text for both the *Lenina* and *IIITH-Indic* databases.

Table 5 displays the F-measures for our heuristic rule based prediction of phrase breaks, for all six languages. An analysis of the numbers shows that, with the exception of Hindi, the rule based system perfroms with F-measures ranging from 0.45 to 0.75.

Language	F-Measure
Telugu(Lenina)	0.62
Telugu(Indic)	0.57
Hindi	0.24
Kannada	0.55
Tamil	0.47
Bengali	0.49

Table 5: F-Measure for rule based prediction of phrase breaks in text

5.2. Phrase break prediction using terminal syllables in a machine learning model

We use the terminal syllable information, as features in a Classification and Regression Tree (CART) framework, to build a model (henceforth referred to as TS model) for predicting phrase breaks from text. As we are using syllable level features in this model, we experiment with different syllable level contexts in order to incorporate contextual information. We use 90% of the text in each language as training data while the remaining 10% was held back for testing. As it is a trivial task to predict breaks at utterance endings, we remove the examples corresponding to utterance ending breaks from the training data.

As an initial experiment, we considered the case where word boundaries that coincide with pauses greater than 80ms are marked as phrase breaks, while all other word boundaries are marked as non breaks. We generated example vectors of both phrase breaks and non breaks, using the terminal syllables along with contextual information. As this was a binary classification task, we also ensured that the number of training vectors of each of the classes (break and non break) were the same. As the number of non breaks were more than the number of breaks in the data, this was achieved by removing examples of non breaks till the total number of example vectors of non breaks and breaks were the same. These example vectors were then used to train the CART model.

We also experiment with different pause thresholds for marking phrase breaks, keeping the context the same in all cases. For the purpose of this experiment we consider the contextual information provided by the previous two syllables and the next two syllables immediately adjacent to the terminal syllable. As before, we take 90% of the text in each language as training data while the remaining 10% was held back for testing and breaks corresponding to utterance endings were omitted from the training data. In this case also, we ensured that the number of training vectors of both classes (breaks and non breaks) were the same.

Language	TS	POS	-1C, TS	-1C, POS	-1C, TS, +1C	-1C, POS, +1C	-2C, TS	-2C, POS	-2C, TS, +1C	-2C, POS, +1C	-2C, TS, +2C	-2C, POS, +2C
Telugu(Lenina)	0.63	0.62	0.69	0.63	0.72	0.69	0.69	0.64	0.72	0.75	0.74	0.75
Telugu(Indic)	0.47	0.56	0.49	0.56	0.53	0.59	0.47	0.52	0.53	0.60	0.52	0.59
Hindi	0.09	0.10	0.09	0.10	0.09	0.09	0.09	0.09	0.09	0.15	0.06	0.18
Kannada	0.37	0.41	0.40	0.42	0.40	0.41	0.39	0.42	0.40	0.44	0.39	0.43
Tamil	0.43	0.39	0.42	0.43	0.39	0.44	0.43	0.44	0.41	0.47	0.42	0.45
Bengali	0.41	0.56	0.34	0.53	0.47	0.53	0.37	0.53	0.49	0.56	0.49	0.54

Table 6: F-Measures for different contexts and setting SSIL $>$ 80ms for phrase breaks, for all six languages where -1C and -2C represents one and two units context respectively to the left and +1C and +2C represents one and two units context respectively to the right

Language	SSIL >50 ms taken as breaks		SSIL >25 ms taken as breaks		all SSIL taken as breaks	
	-2C, TS, +2C	-2C, POS, +2C	-2C, TS, +2C	-2C, POS, +2C	-2C, TS, +2C	-2C, POS, +2C
Telugu(Lenina)	0.72	0.75	0.73	0.75	0.73	0.75
Telugu (Indic)	0.54	0.60	0.54	0.62	0.54	0.62
Hindi	0.37	0.24	0.38	0.28	0.47	0.31
Kannada	0.46	0.51	0.47	0.50	0.55	0.59
Tamil	0.46	0.54	0.55	0.57	0.58	0.61
Bengali	0.51	0.59	0.53	0.59	0.53	0.58

Table 7: F-Measure for different silence thresholds, for all six languages where -2C represents two units context to the left and +2C represents two units context to the right

5.3. Phrase break prediction using POS tag sequence in a machine learning model

We use the same experimental setup as in Section 5.2 changing only the features used. We use the POS tag sequence information, as features in a Classification and Regression Tree (CART) framework, to build a model (henceforth referred to as POS model) for predicting phrase breaks from text. As the POS tag sequence information is a word level feature, we experiment with different word level contexts in order to incorporate contextual information. All the experiments in Section 5.2 are repeated using the POS tag sequence information as features

5.4. Analysis of results

Table 6 shows the performance of both the TS model and POS model, in predicting phrase breaks from text, for all six languages, when word boundaries greater than 80ms are marked as phrase breaks. An observation of the table shows that the performance of both the models for Hindi is poor. In this case, as observed in Section 2.1 the majority of the pause durations are less than 80ms. As as result the number of example vectors for phrase breaks are very few, resulting in a poorly trained model. Also in case of Hindi the sentences are short, and so there are few pauses in the middle of the sentences.

Table 7 shows the prediction accuracy for different pause thresholds for marking phrase breaks, keeping the context same in all cases.

An analysis of the F-measure numbers obtained from all experiments shows that, for automatic prediction of phrase breaks from text, models built using terminal syllables perform nearly as well as models built using traditional features like part-of-speech (POS) sequences.

	% Preference
No Phrasing model	10%
POS model	75%
No Preference	15%

Table 8: AB Test Results for No Phrasing model vs POS model

	% Preference
No Phrasing model	12%
TS model	73%
No Preference	15%

Table 9: AB Test Results for No Phrasing model vs TS model

6. Subjective evaluation of phrasing models

We perform subjective listening tests for Telugu, to compare utterances synthesized by incorporating the TS model and POS model with utterances synthesized with no explicit phrasing model. The listening tests were set up as an ABX task, for native speakers of Telugu. Two phrasing models were compared at a time. An utterance was synthesized by incorporating both models and both versions were presented to the participants in a randomized order, and the participants were asked to mark the version they preferred. They also had an option of no preference if they could not pick one utterance over the other.

In the first listening task, we compared utterances synthesized by incorporating the POS model with utterances synthe-

	% Preference
POS model	35%
TS model	34%
No Preference	31%

Table 10: AB Test Results for POS model vs TS model

sized with no explict phrasing model. For the second listening task, we compared utterances synthesized by incorporating the TS model with utterances synthesized with no explicit phrasing. Finally, we performed a third listening task where we compared utterances synthesized by using the POS model with utterances synthesized using the TS model. All the listening tasks were performed by 10 native speakers of Telugu, who evaluated 15 samples picked randomly from the test set. Tables 8, 9 and 10 show the results of these listening tests.

An examination of the results in Tables 8 and 9 shows that perceptually there is a marked preference for utterances synthesized with both the POS model and the TS model over utterances synthesized with no explicit phrasing. Table 10 shows that when the POS model and the TS model are compared with each other, perceptually there is no significant preference for one model over the other.

7. Conclusions

In this paper we describe phrase break prediction for Text-to-Speech systems in Indian languages. We look at syllable level units and explore the use of terminal syllables to model phrase breaks. We demonstrate the correlation between these terminal syllables and the acoustic breaks found in the speech signal. We also demonstrate that there is a nonlinear relationship between syntax and prosody, and that there are significant variations between syntactic breaks and phrase breaks.

We utilize these terminal syllables in building models for phrase break prediction from text in six Indian languages and demonstrate by means of objective and subjective measures that models built using these terminal syllables perform as well as traditional models built using part-of-speech (POS) sequence information.

The advantage of these terminal syllables, is that they can be directly derived from the text under consideration, thus eliminating the need for additional linguistic resources like shallow parsers or POS taggers, while also eliminating the need to model phrase breaks by computationally expensive unsupervised models.

The samples used for the listening tests are available online at `http://ravi.iiit.ac.in/~speech/SSW8/samples.html`.

In the future we wish to explore the use of Amazon Mechanical turk (MTurk) to conduct the listening evaluations. This would enable us to be able to conduct listening tests with more number of subjects to evaluate the models.

8. Acknowledgements

This work is partially supported by MCIT-TTS consortium project funded by MCIT, Government of India. We gratefully acknowledge the contributions of Prof. Hema Murthy, IIT Madras for the fruitful discussions on the subject. The authors would also like to thank all the volunteers who participated in the perceptual evaluations.

9. References

[1] P. Taylor and A. W. Black, "Assigning phrase breaks from part-of-speech sequences," *Computer Speech and Language*, vol. 12, pp. 99–117, 1998.

[2] K. Prahallad, E. V. Raghavendra, and A. W. Black, "Learning speaker-specific phrase breaks for text-to-speech systems," in *Proceedings of ISCA Speech Synthesis Workshop (SSW7)*, Kyoto, Japan, 2010, pp. 148–153.

[3] K. Prahallad, E. V. Raghavendra, and A. W. Black, "Semi-supervised learning of acoustic driven prosodic phrase breaks for text-to-speech systems," in *Proceedings of 5th International Conference on Speech Prosody (Speech Prosody 2010)*, Chicago, Illinois, May 2010.

[4] C. Wightman, S. Shattuck-Hufnagel, M. Ostendorf, and P. J. Price, "Segmental durations in the vicinity of prosodic phrase boundaries," *Journal Acoustical Society of America*, vol. 91, no. 3, pp. 1707–1717, 1992.

[5] L. Redi and S. Shattuck-Hufnagel, "Variation in realization of glottalization in normal speakers," *Journal of Phonetics*, vol. 29, pp. 407–429, 2001.

[6] H. Kim, T. Yoon, J. Cole, and M. Hasegawa-Johnson, "Acoustic differentiation of l- and l-l% in switchboard and radio news speech," in *Proceedings of Speech Prosody*, Dresden, 2006.

[7] A. Parlikar and A. W. Black, "A grammar based approach to style specific phrase prediction," in *Proceedings of Interspeech*, Florence, Italy, August 2011, pp. 2149–2152.

[8] A. Parlikar and A. W. Black, "Data-driven phrasing for speech synthesis in low-resource langauges," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, Kyoto, Japan, March 2012.

[9] M. Wang and J. Hirschberg, "Automatic classification of intonational phrase boundaries," *Computer Speech and Language*, vol. 6, pp. 175–196, 1992.

[10] E. Navas, I. Hernez, and I. Sainz, "Evaluation of automatic break insertion for an agglutinative and inflected language," *Speech Communication*, vol. 50, no. 11-12, pp. 888–899, 2008.

[11] H. Schmid and M. Atterer, "New statistical methods for phrase break prediction," in *Proceedings of 20th international conference on Computational Linguistics,COLING '04*, Stroudsburg, PA, USA, 2004.

[12] A. Bonafonte and P. Agüero, "Phrase break prediction using a finite state transducer," in *Proceedings of 11th International Workshop on Advances in Speech Technology*, 2004.

[13] B. Busser, W. Daelemans, and A. van den Bosch, "Predicting phrase breaks with memory-based learning," in *Proceedings of 4th ISCA Speech Synthesis Workshop*, 2001.

[14] H. Ney, U. Essen, and R. Kneser, "On structuring probabilistic dependences in stochastic language modeling," *Computer Speech and Language*, vol. 8, pp. 1–38, 1994.

[15] N. S. Krishna and H. A. Murthy, "A new prosodic phrasing model for Indian language Telugu," in *INTERSPEECH-2004-ICSLP*, vol. 1, Oct 6-11 2004, pp. 793–796.

[16] A. Bellur, K. Narayan, K. Krishnan, and H. Murthy, "Prosody modeling for syllable-based concatenative speech synthesis of Hindi and Tamil," in *Proceedings of 2011 National Conference on Communications*, 2011, pp. 1–5.

[17] S. C. Pammi and K. Prahallad, "POS tagging and chunking using decision forests," in *Proceedings of the IJCAI-07 workshop on Shallow Parsing in South Asian Languages*, Hyderabad, India, 2007.

[18] K. Prahallad, E. N. Kumar, V. Keri, S. Rajendran, and A. W. Black, "The IIIT-H Indic Speech Databases," in *Proceedings of Interspeech*, Portland, Oregon, USA, 2012.

[19] K. Prahallad, A. W. Black, and R. Mosur, "Sub-phonetic modeling for capturing pronunciation variations for conversational speech synthesis," in *Proccedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, Toulouse, France, 2006, pp. 853–856.

[20] P. Avinesh and K. Gali, "Part-of-Speech tagging and chunking using conditional random fields and transformation based learning," in *Proceedings of the IJCAI-07 workshop on Shallow Parsing in South Asian Languages*, 2007.

[21] C. J. van Rijsbergen, *Information Retrieval*. Butterworth, 1979.

The Effect of Age and Native Speaker Status on Intelligibility

Catherine Watson[1], *Wei Liu*[1], *Bruce MacDonald*[1]

[1] Department of Electrical and Computer Engineering, University of Auckland, New Zealand
c.watson@auckland.ac.nz,b.macdonald@auckland.ac.nz

Abstract

We investigate whether listener age or native speaker status has the biggest impact on the intelligibility of a synthetic New Zealand English voice. The paper presents findings from a speech intelligibility experiment based on a reminding task involving 67 participants. There were no significant differences in the results due to age (young and old adults), however there was for native speaker status. The non-native listeners performed significantly worse than the native listeners in the synthetic speech condition although no differences were found in the natural speech condition. We argue that despite the fact that aging impacts on speech perception, the older native listeners were able to draw on their in depth language model to help them parse the synthetic speech. The non-native speakers do not have such an in depth model to assist them.

Index Terms: speech synthesis intelligibility, older listeners, non-native listeners.

1. Introduction

Speech is increasingly used as an interface between humans and machines, particularly by social and social assistive robots ([4-14]). One such example is the HealthBots project which has developed social assistive robots for older adults ([10-13]). These robots have been deployed extensively in a New Zealand retirement village. They provide a broad range of services to support the daily activities of the users, including vital signs measurement (e.g. pulse rate, blood pressure and glucose level), schedule reminding (medication, and meetings), telepresence communication, human-robot interaction (e.g. greeting, authentication), and entertainment in the form of jokes and music. The robot communicates to users via a computer screen and a synthetic voice, and the users communicate via buttons on a touch screen.

There is a significant body of work on human-robot interfaces, particularly targeted at the aged population (see for example [7-14]). Acceptance of these healthcare robots by users is crucial for the successful deployment of robotics for aged care. In the HealthBots project, three extensive trials on the feasibility and quality of life effects of the robots in the retirement village have been performed over the last three years (see for example [11-13]). It was found that participants rated the robot highly in terms of overall quality of Human Robot interfaces [12], suggesting participants found the robot acceptable. There were also improvements in attitude towards the robots after meeting the robot [12] and the robot voice plays an important part in its acceptability [17].

1.1. The New Zealand English Voice

A diphone-based male New Zealand English (NZE) voice, running with the Festival Speech Synthesis framework [15] has been developed for the robots [16-19]. An NZE pronunciation dictionary was developed in conjunction with this voice, which contains not only words specific to the NZE lexicon (such as many Maori loan words), but also NZE specific pronunciations (for example the merger of all /iə/ and /eə/ diphthones to /eə/) [16]. The choice of using a New Zealand accent was a deliberate one. People infer much about a person from their accent (e.g. likeability, authority, trustworthiness)[21-23], and negative perception of a voice can cause confusion in communication intent [23]. In socially interactive technologies there is also evidence that the accent influences the impression and acceptance of the interface [eg 17,23,24]. In a New Zealand based study we found people reacted significantly more positively to the robot if it had our NZE voice as opposed to the US accented KAL voice from Festival [17] and in addition they thought the performance of the robot was better. This was despite the fact the task and dialogue the robot performed was exactly the same, only the voices differed. In [18] we specifically tested the perception of speech quality of the NZE voice by older people in a New Zealand retirement village. We found the NZE voice was ranked higher than the KAL voice, and participants reported that they felt they could listen to it longer.

Through speech quality tests we have received feedback on how pleasant people found the NZE robot voice [16-19] and anecdotally via comments collected from the robot trials. Age does effect the perception of quality, with the older listeners giving the voice lower speech quality scores than the younger listeners [18]. These speech quality tests have helped inform us on what improvements are needed for the voice, particularly for the older users. For instance we created a new Festival function that enables different vocal affect, in response to the request to make the voice less robotic [19]. However, there has not yet been a rigorous test of the intelligibility of the NZE voice. The speech is clearly intelligible as the robots have been successfully employed in the trials. But it is important to quantify the intelligibility to identify what areas of the synthetic voice need improvements.

There are user specific issues associated with speech intelligibility for the HealthBots robot. Speech intelligibility, whether it be synthetic or natural, is dependent on factors such as hearing status, listeners age, native speaker status (e.g [25-37]). These latter two are of specific importance to the HealthBots project due to the language background of the two the main user groups of our healthcare robot. These two groups are New Zealand senior citizens (who are currently predominantly NZE speakers), and their carers, many of whom are second language speakers of English [38]. There may be speech intelligibility issues associated with both groups, and we need to have a better understanding about these.

. This paper outlines our investigation into the impact that both age and native speaker status have on speech intelligibility for both the synthetic NZE voice, and a natural voice. In section 1.2 we give a brief overview of other intelligibility studies with relevant findings to our study here. In Section 2 we present the findings of our own study, and discuss the implications of them in Section 3, before concluding in Section 4.

1.2. Speech intelligibility

There is strong evidence that the aging process impacts on speech perception, whether it be the ability to resolve differences in segmental temporal cues (e.g.[25]), understanding speech in background noise (eg. [26]), or, due to degradation in process resourcing (eg. [33]). These difficulties are expected to be further compounded in synthetic speech [27-30]. Studies consistently show that natural speech is more intelligible than synthetic speech [eg. 29-32]. Further, older participants perform consistently worse in intelligibility tests with synthetic speech, compared to younger adults [29-33]. In particular Wolters and colleagues [eg 29,30] demonstrated that older participants make more errors compared to younger participants in recalling complex words, such as medicine names.

However, regardless of age, Native speakers do have a vast knowledge of the sound system of their language, and for degraded speech they are able to exploit the fact there is a large amount of redundancy in spoken language, which is in contrast to the non-native speaker [28,36,37]. Being a non-native speaker is known to impact on intelligibility of synthetic speech [27,32,34], although the level of skill in the host language can reduce the impact [27]. However it is of note that Jones et al [28] found no difference in the comprehension rates between native and non-native speakers of English for a synthetic Australian English voice.

To date there has been no study that contrasts the synthetic speech intelligibility scores by listeners who are older adults to those who are non-native. In [32] they found older listeners performed worse than younger listeners in a synthetic speech intelligibility task. When they then split their young listeners group into native and non-native speakers, they found the non-native speakers performed worst. Perhaps the older first language listeners would have performed worse than younger non-native listeners, but this was not tested.

In this study we will explicitly look at synthetic speech intelligibility issues contrasting older and younger listeners, and native and non-native listeners. We expect greater accuracy with stimuli from natural speech than synthetic speech, and from less complex stimuli than complex stimuli. We also expect that the older listeners would find the intelligibility task harder than the younger listeners , and the non-native speaker listeners to find the task harder than native speaker listeners.

2. Speech Intelligibility Study.

We based our Speech Intelligibility study on a schedule reminding task proposed by Wolters et al ([29,30]). This tasked involved reminders about meeting people, and taking medicines. This experiment was ideally suited to our study, as reminding is one of the main services of the HealthBot Robots. In the exercise, participants were given a reminder about either meeting a person or taking medication at a specified time. They then were then required to identify the time the action was to take place, the person's name, or the medication name. For this study the robots were not involved. Participants only heard the voice via an online survey. We did try to recruit participants from the retirement village, where the robots were being tested, however this proved very difficult. This survey came immediately after the three robot trials and the residents most likely had experiment fatigue. By having the survey online, we were able to recruit participants beyond the retirement village.

2.1. Methodology

In the study participants were given reminders in either a natural NZ voice or the synthetic NZE voice and we compared participant's abilities to correctly recall the reminders. Approval was granted for this study by The University of Auckland Human Participation Ethics Committee.

2.1.1. The stimuli

The reminders the participants were given are listed in a generalized form in Table 1. Table 2 lists the times, people names, and medicine names used in the recall task. For the times we state the time of the day (morning, afternoon, evening), rather than stating AM and PM as in [29,30] as we believe this is more natural in spoken speech. In addition the names of the people and medicines were also adapted from [29,30] to be more applicable to New Zealand.

Table 1 The general form of the reminder stimuli. These were first proposed in [29,30].

Reminder	Template
Meeting Reminder	At "time", you are meeting PERSON. You are meeting "person" at TIME.
Medication Reminder	At "time", you need to take your MEDICATION. You need to take your MEDICATION at "time".

Table 2 The times, names of people, and names of medicines used in this study, adapted from [29,30].

Category	Stimuli
TIME	5:05 in the afternoon, 2:25 in the afternoon, 8:45 in the afternoon, 6:40 in the evening, 11:40 in the morning, 8:20 in the morning, 8:10 in the evening, and 2:35 in the afternoon.
PERSON	Judy, Julie, Nicky, Ricky, Kim, Jim, Ted, and Ned
MEDICATION	Accumycin, Omeprozole, Beclotor, Cilazapril, Colecalciferol, Dexozine, Digoxin and Felodopine.

The focus items for this study are capitalized in Table 2. Participants would hear a reminder, and then were required to respond to a questions about the reminder (e.g. for the reminder "You are meeting Jim at 5:05 in the afternoon", they would be asked: "What time will you meet Jim?". For the synthetic NZE voice the focus word was emphasized by placing a small pause immediately before to it, along with a slight pitch rise syllable prior to the focus word.

For both voices the questions were formulated so the focus item was usually in the second part of the sentence, but for medication names we also asked participants to recall these when they fell in the first part of the sentence. In a preliminary study [35] we found, as in [29,30], that participants recalled items better if they occurred in the second part of the sentence in the reminder. There was no need, therefore, to retest this thoroughly in the larger study.

There were 32 reminders in total: 8 TIME and NAME reminders, and 16 MEDICINE reminders. The order of the reminders was randomized. Two versions of these 32 reminders were compiled; one with natural speech (also an NZE male speaker), and one with the NZE voice. The text content was exactly the same.

2.1.2. The web-based survey

The entire intelligibility test was delivered online using the open source software LimeSurvey (www.limesurvey.com) on a web server at The University of Auckland. Participants were required to have a playback facility on their computer. At the beginning of the survey, the web service provided participants with a client-system test to ensure that both the survey website and the multimedia player were running correctly on the user's PC. Prior to the Speech intelligibility task, participants were given an example question and answer. Participants typed their response to each reminder into a text box. They were only able to listen to each reminder once. The intelligibility part of the survey took about 20 minutes to complete. There was a speech quality survey as well [35]. This took about 10 minutes complete and was presented first. Only the results of the intelligibility test are presented here.

2.1.3. Participants

Eighty-one participants were recruited for the experiment. Unfortunately 12 participants only partially completed it. Most likely this was due to either participant fatigue or issues with the internet connection, judging by their comments. Of the remaining 69 participants 50 were in the young group, aged between 18-25 years, and 19 were in the older group, aged over 45 years. The majority the older group was aged over 60 (12/19). In addition 43/69 participants had English as their first language (henceforth referred to as the L1 group), and 26 were second language speakers of English (henceforth referred to as the L2 group). Twenty-four of the L2 group were from the young group. They had lived in an English-speaking country for a mean of 9.8 years, with a standard deviation of 4.8 years. The remaining two L2 participants were in the older group. However we removed these two from the experiment, as this wasn't a sufficiently high enough number of participants for the old L2 category to be experimentally viable. The very low number in the old L2 group was due to the fact we did not specifically target this group, but the online survey was open to everyone.

Given the task was online; it was not feasible to do hearing tests, however participants did have to fill out a questionnaire about themselves, which included a question on hearing. Four of the older group reported hearing problems, none of the younger group did. The participants heard the reminders being given in one voice type only, and were randomly allocated to one of the two categories (natural or synthetic). Table 3 below lists the age, language background and number of participants hearing each voice.

Table 3 the number of participants who heard each voice type, according to age and language experience.

Voice Used In Task	Young		Old
	L1	L2	L1
Natural	10	9	4
NZE voice (NZE)	16	15	12

2.1.4. Analysis of Data

All responses were scrutinised and compared to the intended word. It was necessary to use a certain amount of discretion to decide whether a response was correct or not. Participants were only asked to write down what they heard and were assured that it was not necessary to spell the word exactly right. There was very little variation in the spelling of people's names, but there was for the spelling of the medicine, and any reasonable phonetic spelling was accepted (for e.g. for the Drug Accumycin acceptable answers included "Ackumisin", "Acumysene", "Accumycin", unacceptable answers included "Attemeisin" "Aclimentin: "accumice"). All times where entered in a digit format and an indication of the time of day (e.g. 5:05 pm). All responses were marked as correct or incorrect, and then the package R [39] was used to do the statistical analysis.

2.2. Results

A multi-way ANOVA was used to analyse the data. The number of correct responses was the dependent variable, and there were three independent variables: participant type (older L1 adult , younger L1 adult, younger L2 adult), voice (natural vs. synthetic), and Stimuli type (TIME, PERSON's name, MEDICINE). All three independent variables were found to be significant. Posthoc Tukey HSD tests, and t-tests were then used to further examine the results. These are discussed in detail below.

2.2.1. The effect of voice

There were more correct responses for the natural voice (mean score 76.8%, SD 20.7%) than for the synthetic NZE voice (mean score 71.6 %, SD 24.3%), and this was significant ($F(1,261)=4.5, p>0.05$).

2.2.2. The effect of L1 vs L2

There were significant differences in the responses of the three different participant categories ($F(2,261)=11.9$, $p<1e-4$). The difference was an L1 vs L2 effect only. There was no significant difference between the two L1 groups (Posthoc Tukey $p=0.9$), the mean correct score for the old L1 was 77.9% (SD 25.1%), and for the young L1 was 77.1% (SD 19%). In contrast the L2 Young group (mean correct score of 66.0 %, SD=24.1%) performed significantly worse than the L1 Young group (Posthoc Tukey $p <1e-3$), and the L1 Old group (Posthoc Tukey $p<1e-3$). For age related effects it is only appropriate to compare the two difference L1 groups, and for this study there is no significant difference. Although it is interesting that the L1 old group was more variable than the young L1 group.

Table 4 L1 and L2 groups Mean Correct scores for all the stimuli (*left*),NZE voice(*middle*),Natural (*end*).

	Total	NZE	Natural
L1 group	77.5%	76.6%	79.2%
L2 group	66.0%	61.0%	72.9%

The mean correct scores for the L1 and L2 participants for all the stimuli, both Natural and Synthetic, are given in Table 4. The results of the young and old L1 groups have been pooled together since there was no significant difference between the two. The mean correct scores from the Natural speech are much higher than for the synthetic speech, which is expected since the Voice factor was significant. Interestingly although the L2 group performed worse than the L1 group for both the Natural and Synthetic conditions, only the differences in the Synthetic condition are significant. Posthoc t-test (Bonferoni corrected for multiple comparisons) showed that the L2 responses for the synthetic stimuli were less than both those from the young L1 group ($t[112.5]=3.8, p<1e-2$), and the old L1 ($t[104.9]=3.0, p<0.05$). In addition the t-tests showed that there were no differences between any of the three groups in the natural speech conditions.

2.2.3. The effect of Stimuli

Table 5 Results of the intelligibility of different stimuli in a sentence

Stimuli	Overall	L1	L2
FM	59.7%	65.1%	50.0%
SM	62.9%	66.0%	57.3%
Name	87.1%	91.9%	78.6%
Time	83.8%	86.9%	78.1%

The stimuli type also impacted significantly on intelligibility, as can be seen in Table 5 ($F(3,261)=36.8, p=0$), where FM stands for the medication name in the first, or earlier, position in the sentence, and SM stands for the MEDICATION name in the second, or latter, position. All of the NAME and TIME stimuli were located in the second position in the sentence. Posthoc Tukey HSD tests on the overall results showed that both the NAME and TIME stimuli have a significant higher intelligibility than medication in either position in a sentence (NAME–FM: diff = 27.4%, p = 0; SM–NAME: diff = -24.3, p = 0; TIME–FM: diff = 24.1%, p = 0; TIME–SM: diff = 20.9%, p = 0). There is no notable difference between the intelligibility of MEDICATION stimuli located in either position in the sentence (SM–FM: diff = 3.2, p = 0.77), nor any intelligibility differences between TIME and NAME stimuli (TIME–NAME: diff = -3.4, p = 0.74).

From our earlier findings (see section 2.2.2) we would expect the L2 participants to do worse than the L1 participants. This can be seen clearly in Table 5, across all 4 stimuli groups. However there was no significant interactions between the stimuli type and language background of the participants, which means the tasks the L1 listeners found easy were also found to be easy by the L2 listeners (i.e. identifying the names and times), and the tasks the L1 listeners found hard, were also found hard by the L2 listeners (i.e. identifying medicine names).

3. Discussion

Some of the findings of this study reinforced our earlier expectations, however some were unexpected. As expected the participants recalled items better if they heard the reminders in the natural voice, as opposed to the synthetic voice. This was also found in [29-31,33]. We note though, that there was only an absolute difference of 5% in the overall intelligibility between the natural and synthetic speech. We also found that less complex stimuli (in our case times and peoples' names) were much easier to recall than the complex ones (in our case medicine names), this was also found in [29-31]. However in contrast to [29,30], we found there was no significant difference in the ability to recall a medicine name regardless of its position in a sentence. It is possible that our method of increasing focus on the keyword (see Section 2.1.1) may have aided in reducing the impact of sentence position on the keyword.

The most significant results from this study are that native speaker status has a greater impact on the speech intelligibility than age. In fact the old L1 speakers performed at the same level as the young L1 speakers, this also is unexpected.

One possible explanation for the high performance of the old L1 group was they were not old enough. The oldest L1 group in this study comprised 4 in the over 75 years group and 8 in the 60-75 years group, however there were also 7 between 45-60 years (although most in this group were nearly 60 years). However when we redid the ANOVA, splitting the L1 group into three groups (the young L1 speakers (as before), those between 45-60 years (7 in this group), and those 60 years and over (12 in this group)) the same variables remained significant. Further, there remained no significant difference between the young L1 speakers (77.1%) and old L1 group - now aged over 60 years old (mean 73.4%).

Other studies have indicated that it is the hearing status of the participants, not the chronological age that is the most important influence on performance [eg. 29,30,33]. The over 60 years L1 group, had all the self–reported cases of hearing impairment, and yet their intelligibility scores were still not significantly different from the young L1, although their mean score is less than the young L1 speakers. Perhaps by increasing the participant numbers this difference would become significant. However the relatively good performance of the older group is further evidence that our methods employed to improve the intelligibility of the voice are working. As is the fact that no significant differences was found for the L1 speakers between their speech intelligibility scores for the synthetic NZE voice and the natural voice (see section 2.2.2).

It is also very noteworthy that there was no significant difference between the L1 and L2 group in the natural voice condition (with a mean of 79.2% and 72.9% respectively, see Table 4). The L2 group only perform significantly worse than the L1 group in the synthetic speech condition (mean 61.9% vs 76.6%, see Table 4). This suggests that it is not the complexity of the reminding task, in particular the spelling of the medicine names that is causing the comparative low score for the L2 group, it is the processing the synthetic speech.

Previous speech intelligibility studies have attributed the poorer performance of the L2 participants, compared to the L1 participants, to less depth of knowledge in both the sound system and linguistic structure of their second language [27, 28, 36]. By contrast the poor performance of older participants compared to young participants in intelligibility studies has been attributed to hearing issues [eg. 29,30,33] and memory issues [eg. 29,30]. However, as noted by [28,36, 37] adult L1 listeners have an in depth knowledge of the sound system and linguistic structure of their language. This helps with intelligibility in natural speech; in [36] they showed L1 listeners benefit from clear speech, but L2 listeners do not. Our study here suggests that whilst the synthetic speech increases the cognitive load on the listeners, the L1 listeners are able to draw on their language knowledge in a way the L2 cannot. Thus whilst the older L1 participants would have all

had age related hearing loss [26] (some to the level it was self-reported) their in depth language knowledge helps compensate for this. A natural conclusion from this study would be that a group of older L2 participants would perform the worst at this task. As we only had two participants in this category, we are not yet in a position to test this, once again more participants would be required.

However for the HealthBots project the findings from this experiment are significant. The numbers of overseas born careers for the New Zealand Elderly have been steadily increasing, between 1991 and 2006 the numbers of careers born overseas increased from 19% to 25% [38], and this increase continues [40]. But more significantly the number of careers for whom English was not their first language has also increased dramatically [40].Therefore the fact that the young L2 group performed significantly worse than either of the L1 groups in our study has significant implications for human-machine interfaces in the HealthBots project, and in fact for all human machine interfaces in healthcare of the Elderly. This is because throughout the western world there is an increasing reliance on carers whose first language is not the language of the country in which they live [38].

In the HealthBots project the focus to date has been improving the intelligibility of our Healthcare robot voice for the Elderly users. We knew that speech intelligibility of the robot would also be an issue for their carers (many of whom have English as a second language), however we made the assumption that age would be a bigger disadvantage than language background. The result of this study suggest this assumption, has been wrong. Therefore although issues with the elderly population remain our main focus, it is now clear we need to also perform our intelligibility tests on L2 speakers too.

4. Conclusions

In this study we have shown that in a reminder task complex words are harder to recall than simple words, reinforcing the findings of [29-31]. The intelligibility of the NZE voice compares favourably to natural speech; in fact for the First Language listeners regardless of age, there was no significant difference between the intelligibility of the two voice types. However for the Second Language listeners the NZE voice is significantly less intelligible than the natural voice. Consequently the main findings of this study, is that for the synthetic NZE voice it is the native speaker status of the listener, not age that has the biggest impact on speech intelligibility. This has major implications for human-robot interfaces in healthcare owing to large number of increasing number of carers in the health systems who are second language speakers in their country of employment.

5. Acknowledgements

We thank the participants of the study, the HealthBots project members, and the New Zealand Ministry for Science and Innovation for funding the HealthBots project. We also thank ETRI for their contributions to the HealthBots project.

6. References

[1] Murray, M. K, "The Nursing Shortage: Past, Present, and Future,"Journal of Nursing Administration, vol. 32, p. 79, 2002.

[2] Super, N., "Who will be there to care? The growing gap between caregiver supply and demand," in National Health Policy Forum, George Washington University, Washington DC, 2002.

[3] United Nations, World Population Prospects: The 2006 revision. New York: United Nations, 2006.

[4] Ichbiah, D., Robots: From science fiction to technological revolution. New York: Harry N Abrams, 2005.

[5] Krebs, H.I., Palazzolo, J.J Dipietro, L.,Ferraro, M., Krol, J., Rannekleiv, K.,Volpe, B.T and Hogan, N. "Rehabilitation robotics: Performance-based progressive robot-assisted therapy," Autonomous Robots, vol. 15, pp. 7-20, 2003.

[6] Granata, C., Chetouani, M., Tapus, A., Bidaud P.and Dupourqué, V., "Voice and graphical-based interfaces for interaction with a robot dedicated to elderly and people with cognitive disorders", in Proc. of the 19th IEEE International Symposium on Robot and Human Interactive Communication, pp.785–790, 2010.

[7] Mataric, M.J., Eriksson, J., Feil-Seifer, D.J. and Winstein, C.J., "Socially assistivee robotics for post-stroke rehabilitation," J. of NeuroEngineering and Rehabilitation, vol. 4, p. 5, 2007.

[8] Ellison, L.M, Pinto, P.A, Kim, F., Ong, A.M, Patriciu, A., Stoianovici, D., Rubin, H., Jarrett, T., and Kavoussi, L.R., "Telerounding and patient satisfaction after surgery," Journal of the American College of Surgeons, vol. 199, pp. 523-530, 2004.

[9] Pollack, M., Engberg, S., Matthews, J.T., Thrun, S., Brown, L., Colbry, D., Orosz, C., Peinter, B., Ramakrishnan, S.,Dunbar-Jacob, J., McCarthy, C., Montemerlo, M., Pineau, J., and Roy, N., "Pearl: A Mobile Robotic Assistant for the Elderly," Workshop on Automation as Caregiver: the Role of Intelligent Technology in Elder Care, (AAAI), August 2002.

[10] MacDonald, B., Abdulla, W., Broadbent, E., Connolly, M., Day, K., Kerse, N., Neve, M., Warren, J., and Watson, C.I., "Robot assistant for care of older people," in Proceedings from the 5th International Conference on Ubiquitous Robots and Ambient Intelligence, 20–22 November 2008.

[11] Jayawardena, C., Kuo, I.H., Unger, U., Igic, A., Wong, R., Watson, C.I., Stafford, R.Q., Broadbent, E., Tiwari, P., Warren, J., Sohn, J., MacDonald, B.A., "Deployment of a service robot to help older people," in Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots & Systems (IROS), pp.5990-5995, Oct. 2010.

[12] Stafford, R.Q, Broadbent, E., Jayawardena, C.,Unger, U., Kuo, I.H., Igic, A., Wong, R., Kerse, N., Watson, C.I., MacDonald, B.A, "Improved robot attitudes and emotions at a retirement home after meeting a robot,"2010 IEEE RO-MAN, pp.82 – 87, Sept. 2010.

[13] Jayawardena, C., Kuo, I., Datta, C., Stafford, R.Q., Broadbent, E., and Macdonald, B.A., "Design, implementation and field tests of a socially assistive robot for the elderly: HealthBot Version2" 4th IEEE RAS/EMBS Int. Conf. on Biomedical Robotics and Biomechatronics Roma, Italy. Pp1837-1842, 2012.

[14] Giuliani, M.V., Scopelliti, M and Fornara, F. "Elderly people at home: technological help in everyday activities," in IEEE international workshop on robots and human interactive communication, USA, 2005, pp. 365-370.

[15] Black, A., Taylor, P., and Caley, R., "The Festival speech synthesis system," 1998. Online: http://www.cstr.ed.ac.uk/projects/festival.html.

[16] Watson, C.I, Teutenberg, J., Thompson, L., Roehling, S., and Igic, A., "How to build a New Zealand voice," in NZ Linguistic Society Conference, Palmerston North, Nov30 – Dec 1 2009

[17] Tamagawa, R. Watson, C.I., Kuo, I.H., Macdonald, B.A., and Broadbent, E., "The Effects of Synthesized Voice Accents on User Perceptions of Robots," International Journal of Social Robots, vol. 3, no. 3, pp. 253–262, Aug. 2011.

[18] Igic, A., Watson, C.I., Macdonald, B.A, Broadbent, E. , Jayawarden, C.J, and Stafford, R., "Perception of Synthetic Speech with Emotion Modeling Delivered through a Robot Platform: An Initial Investigation with Older Listeners", in The Proceedings of the 13th Australasian International Conference on Speech Science and Technology, pp. 189–192, 2010.

[19] Igic, A., Watson, C.I. Teutenberg, J.D, Tamagawa, R., MacDonald, B.A., and Broadbent, E.,"Towards a Flexible Platform for Voice Accent and Expression Selection on Healthcare Robot", in The Proc. 2009 Australasian Language

Technology Workshop. 7, L. A. Pizzato and R. Schwitter (Eds). Sydney, 3 Dec. 2009–4 Dec. 2009, pp. 109–113, 2009.

[20] Bayard D, "The cultural cringe revisited: changes through time in KIWI Attitudes towards accents." In: Bell A, Kuiper K (eds) New Zealand English. Benjamins, Amsterdam, pp 297–324, 1999.

[21] Bayard D, Weatherall A, Gallois C, Pittam J "Pax Americana? Accent attitudinal evaluations in New Zealand", Australia,and America. J Socioling 5:22–49 , 2001.

[22] Cargile A, Giles H "Understanding language attitudes: exploring listener affect and identity." Lang Commun 17:195–217, 1997.

[23] Walters ML, Syrdal DS, Koay KL, Dautenhahn K, te Boekhorst R "Human approach distance to a mechanical-looking robot with different robot voice styles". In: Proc the 17th IEEE international symposium on robot and human interactive communication, Munich, Germany, pp 707–712, 2008.

[24] Goetz J, Kiesler S, Powers A "Matching robot appearance and behavior to tasks to improve human-robot cooperation". In: Proc the 12th IEEE Int. Symp. on Robot and Human Interactive Communication, Millbrae, California, USA, pp 55–60, 2003.

[25] Lister, J., and Tarver, K.,"Effects of age on silent gap discrimination in synthetic speech stimuli," J. Speech Lang. Hear. Res. 47, 257–268, 2004.

[26] Kim S, Frisina RD, Frisina DR. "Effects of age on speech understanding in normal hearing listeners: relationship between the auditory efferent system and speech intelligibility in noise." Speech Communication Vol. 48 855-862, 2006.

[27] Alamsaoutra, D.M., Kohnert, K.J., Munson, B., Reichle, J., "Synthesized speech intelligibility among native speakers and nonnative speakers of English". Augment. Altern. Commun. 22, 258–268. 2006

[28] Jones, C., Berry, L., and Stevens, C.,"Synthesized speech intelligibility and persuasion: Speech rate and non-native listeners," Computer Speech and Language, vol. 21, no. 4, pp. 641–651, Oct. 2007.

[29] Wolters, M., Campbell, P., DePlacido, C., Liddell, A., and Owens, D., "Making Speech Synthesis More Accessible to Older People." 6th ISCA Wrkshp on Speech Synthesis (SSW-6), 2007.

[30] Wolters, M, Campbell, P., DePlacido, C., Liddell, A., and Owens, D., "The effect of hearing loss on the intelligibility of synthetic speech," in Proceedings of the 16th International Congresss of the ICPHS, pp. 673–676, Aug. 2007.

[31] Humes, L.E., Nelson, K.J., Pisoni, D.B., and Lively, S.E., "Effects of Age on Serial Recall of Natural and Synthetic Speech," Journal of Speech and Hearing Research, vol. 36, pp. 634–639, 1993.

[32] Langner A., and Black, A.W., "Using Speech In Noise toImprove Understandability for Elderly Listeners," in Proceedings of ASRU, San Juan, Puerto Rico, 2005.

[33] Roring, R.,W., Hines,F.G., and Charness, N., "Age Differences in Identifying Words in Synthetic Speech," Human Factors: The Journal of the Human Factors and Ergonomics Society, vol. 49, no. 1, pp. 25–31, Feb. 2007.

[34] Reynolds, M., Bond, Z., Fucci, D., "Synthetic speech intelligibility: comparison of native and non-native speakers of English". Augmentative and Alternative Communication 12, 32–36. 1996.

[35] Liu, W "Assessing and Improving the Intelligibility of synthetic voices on a healthcare robot" University of Auckland. Unpublished Masters Thesis. 2013

[36] Bradlow A.R., and Bent T., "The clear speech effect for non-native listeners" Journal of the Acoustical Society of America 122 (1), 272-284 2002.

[37] Hongyan, W, and van Heuvan, V.F. "Quantifying the Interlanguage Speech Intellibility Benefit." The proceedings of ICPhS XVI , Saarbrücken, 6-10 Aug , pp 1729-1732, 2007,

[38] Badkar, J., P. Callister & R. Didham. (2009). Aging New Zealand: The Growing Reliance on Migrant Caregivers. Wellington: Victoria University.

[39] R Development Core Team (2011). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0, URL http://www.R-project.org/

[40] Maclagan, M. and Grant, Annabel. (2011) Care of people with Alzheimer's Disease in NZ: Supporting the Telling of Life Stories. Peter Backhaus (ed) Communication in Elderly Care: Cross-Cultural Perspectives London: Continuum. P 62-89.

EXEMPLAR-BASED VOICE CONVERSION USING NON-NEGATIVE SPECTROGRAM DECONVOLUTION

Zhizheng Wu[1,2], Tuomas Virtanen[3], Tomi Kinnunen[4], Eng Siong Chng[1,2], Haizhou Li[1,2,5]

[1]School of Computer Engineering, Nanyang Technological University, Singapore
[2]Temasek Laboratories@NTU, Nanyang Technological University, Singapore
[3]Department of Signal Processing, Tampere University of Technology, Tampere, Finland
[4]School of Computing, University of Eastern Finland, Joensuu, Finland
[5]Human Language Technology Department, Institute for Infocomm Research, Singapore

wuzz@ntu.edu.sg

ABSTRACT

In the traditional voice conversion, converted speech is generated using statistical parametric models (for example Gaussian mixture model) whose parameters are estimated from parallel training utterances. A well-known problem of the statistical parametric methods is that statistical average in parameter estimation results in the over-smoothing of the speech parameter trajectories, and thus leads to low conversion quality. Inspired by recent success of so-called exemplar-based methods in robust speech recognition, we propose a voice conversion system based on non-negative spectrogram deconvolution with similar ideas. Exemplars, which are able to capture temporal context, are employed to generate converted speech spectrogram convolutely. The exemplar-based approach is seen as a data-driven, non-parametric approach as an alternative to the traditional parametric approaches to voice conversion. Experiments on VOICES database indicate that the proposed method outperforms the conventional joint density Gaussian mixture model by a wide margin in terms of both objective and subjective evaluations.

Index Terms— Voice conversion, exemplar, non-negative matrix factorization, non-negative matrix deconvolution, temporal information

1. INTRODUCTION

Voice conversion is a process of modifying source speaker's voice to sound like it was spoken by another speaker (target). It can be applied to speaker identity conversion in speech synthesis systems when only a few recording samples from a specific target speaker are available.

In general, voice conversion techniques operate on several different speech features, such as spectral envelope [1, 2], formants [3], fundamental frequency [4, 5] and duration [6]. Spectral envelope contains most of the speaker identity information and is the focus in most of the voice conversion studies, including this one. Spectral conversion involves two phases, training and run-time conversion. During training, a transformation function is estimated from frame-aligned source-target feature vectors. The trained conversion model is then applied to unseen utterances at system run-time. Implementation of the conversion function is the most important part of a voice conversion system.

To implement a robust spectral conversion function, a number of data-driven statistical parametric methods have been proposed in the past two decades. A straightforward way to model the relationship between source and target speech is to employ vector quantization (VQ) to learn a codebook from the paired source-target frame vectors, and apply this codebook during conversion phase [7].

To alleviate the frame-to-frame discontinuity problem caused by VQ, joint density Gaussian mixture model (JD-GMM) was proposed [8, 9, 1]. It implements a smoothed local linear transformation function for each frame. Other local linear transformation methods, such as partial least square regression [10], trajectory GMM/hidden Markov model (HMM) [11], mixture of factor analyzers [12], local linear transformation [13], noisy channel model [2] and so on, have been proposed to reduce the over-smoothing and over-fitting problems of JD-GMM. In addition to the linear transformation functions, which assume the source and target speech features to be linearly correlated, nonlinear methods, such as artificial neural network [3, 14], support vector regression [15], kernel partial least square regression [16], and conditional restricted Boltzmann machine [17], have been studied to implement nonlinear conversion.

Due to inherent statistical averaging in parametric methods, over-smoothed speech samples are generated from the averaged parameters, which leads to unnatural speech quality. Inspired by the success of so-called *exemplar*-based noise robust speech recognition [18, 19, 20], we propose a non-parametric exemplar-based voice conversion method as an alternative to statistical parametric methods. We define an exemplar to be a segment of speech spectrogram spanning multiple frames. Utilizing multiple frames, as opposed to single frame in the conventional methods, allows contextual modelling which helps increasing the resulting speech quality.

We study two examplar-based voice conversion variants: *non-negative spectrogram factorization (NMF)* and *non-negative spectrogram deconvolution (NMD)*. In the former variant, each spectrogram frame is represented as a convex combination of several basis spectra (atoms) forming a dictionary. In the deconvolution variant, a converted spectrogram is generated as a convolution of exemplars and activations. Comparing with the most related work in [21], our work has the following novel contributions:

a) We utilize multiple-frame exemplar rather than single-frame spectrum as the basis in the dictionary;

b) We employ low-dimensional filter-bank energies instead of the original magnitude spectrum to represent source spectrogram and source dictionary for efficient computation;

c) We employ a convolutive model to include temporal context information in the converted spectrogram.

2. BASELINE JOINT DENSITY GAUSSIAN MIXTURE MODEL METHOD

Among the statistical parametric methods, joint density Gaussian mixture model (JD-GMM) method [8, 1] is one of the most successful methods, due to the probabilistic treatment and flexible implementation. Therefore, we employ the JD-GMM method as our baseline method in this study.

The JD-GMM method involves two phases: off-line training and run-time conversion phases. During the training phase, given parallel training data from a source speaker $\mathbf{X}$ and a target speaker $\mathbf{Y}$, dynamic time warping (DTW) algorithm is used to align the source speech vectors and target speech vectors to obtain the paired speech feature vector $\mathbf{Z} = [\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_t, \ldots, \mathbf{z}_T]$, where $\mathbf{z}_t = [\mathbf{x}_n^\top, \mathbf{y}_m^\top]^\top \in \mathcal{R}^{2d}$, and $\mathbf{x}_n \in \mathcal{R}^d$ and $\mathbf{y}_m \in \mathcal{R}^d$ are source and target speech feature vectors, respectively.

Gaussian mixture model (GMM) is adopted to model the distribution of the paired feature vector sequence $\mathbf{Z}$, which represents the joint distribution of source speech $\mathbf{X}$ and target speech $\mathbf{Y}$. The joint probability density is given as follows:

$$P(\mathbf{X}, \mathbf{Y}) = P(\mathbf{Z}) = \sum_{k=1}^{K} w_k^{(z)} \mathcal{N}(\mathbf{z}|\boldsymbol{\mu}_k^{(z)}, \boldsymbol{\Sigma}_k^{(z)}), \quad (1)$$

$$\boldsymbol{\mu}_k^{(z)} = \begin{bmatrix} \boldsymbol{\mu}_k^{(x)} \\ \boldsymbol{\mu}_k^{(y)} \end{bmatrix}, \boldsymbol{\Sigma}_k^{(z)} = \begin{bmatrix} \boldsymbol{\Sigma}_k^{(xx)} & \boldsymbol{\Sigma}_k^{(xy)} \\ \boldsymbol{\Sigma}_k^{(yx)} & \boldsymbol{\Sigma}_k^{(yy)} \end{bmatrix},$$

where K is the number of Gaussian components, $\boldsymbol{\mu}_k^{(z)}$ and $\boldsymbol{\Sigma}_k^{(z)}$ are the mean vector and the covariance matrix of the k^{th} Gaussian component $\mathcal{N}(\mathbf{z}|\boldsymbol{\mu}_k^{(z)}, \boldsymbol{\Sigma}_k^{(z)})$, respectively. The prior probability $w_k^{(z)}$ of the k^{th} Gaussian component is constrained by $\sum_{k=1}^{K} w_k^{(z)} = 1$. To estimate the model parameters of the joint density Gaussian mixture model $\lambda^{(z)} = \{w_k^{(z)}, \mu_k^{(z)}, \boldsymbol{\Sigma}_k^{(z)} | k = 1, 2, \ldots, K\}$, the well-known expectation-maximization (EM) algorithm is adopted to maximize likelihood of the training data.

In the run-time conversion phase, JD-GMM model parameters are employed to implement the conversion function. To be more specific, for each input source speech feature vector $\mathbf{x}$, the conversion function $F(\mathbf{x})$ implemented with minimum mean square error is used to predict the target's feature vector $\hat{\mathbf{y}}$ is given as:

$$\hat{\mathbf{y}} = F(\mathbf{x}) = \sum_{k=1}^{K} p_k(\mathbf{x})(\boldsymbol{\mu}_k^{(y)} + \boldsymbol{\Sigma}_k^{(yx)}(\boldsymbol{\Sigma}_k^{(xx)})^{-1}(\mathbf{x} - \boldsymbol{\mu}_k^{(x)})), \quad (2)$$

$$p_k(\mathbf{x}) = \frac{w_k \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_k^x, \boldsymbol{\Sigma}_k^{xx})}{\sum_{k=1}^{K} w_k \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_k^x, \boldsymbol{\Sigma}_k^{xx})},$$

where $p_k(\mathbf{x})$ is the posterior probability of the source vector $\mathbf{x}$ generated from the k^{th} Gaussian component.

We note that during the JD-GMM model parameter estimation process, the mean vector of each Gaussian component is updated as:

$$\boldsymbol{\mu}_k^{(z)} = \frac{\sum_{t=1}^{T} \mathbf{z}_t p_k(\mathbf{z}_t, \lambda^{(z)})}{\sum_{t=1}^{T} p_k(\mathbf{z}_t, \lambda^{(z)})}. \quad (3)$$

Similarly, the covariance matrix of each Gaussian component is updated as:

$$\boldsymbol{\Sigma}_k^{(z)} = \frac{\sum_{t=1}^{T} p_k(\mathbf{z}_t, \lambda^{(z)})(\mathbf{z}_t - \boldsymbol{\mu}_k^{(z)})(\mathbf{z}_t - \boldsymbol{\mu}_k^{(z)})^\top}{\sum_{t=1}^{T} p_k(\mathbf{z}_t, \lambda^{(z)})} \quad (4)$$

From (3) and (4), we observe that when calculating mean and covariance for each Gaussian component, all the training samples are used, which is the so-called statistical average. The statistical average results in over-smoothing of the converted speech. We also note that if the correlation between the paired source and target feature vectors is low, the value of the covariance matrix $\Sigma_k^{(yx)}$ will be very small, therefore, only $\boldsymbol{\mu}_k^{(y)}$ contributes to the converted speech as observed and reported in [22].

3. PROPOSED EXEMPLAR-BASED VOICE CONVERSION METHOD

To tackle the over-smoothing problem, we propose an exemplar-based method to generate the converted speech from the spectrogram segments (exemplar). We employ two matrix factorization techniques to implement the exemplar-based method: non-negative spectrogram factorization and non-negative spectrogram deconvolution. Both implementations have the same procedures as follows:

1 Training: construct parallel source and target dictionaries;

2 Conversion:

- 2.a Extract source spectrogram;
- 2.b Given source spectrogram and source dictionary, estimate activation matrix;
- 2.c Utilize the activation matrix estimated in step 2.b and the target dictionary to generate the converted spectrogram;

The two implementations using matrix factorization techniques are briefly introduced in this section.

3.1. Non-negative spectrogram factorization (NMF)

The first exemplar-based method is based on *non-negative spectrogram factorization*. The basic idea of this method is to represent a magnitude spectrum as a linear combination of a set of basis spectra (*speech atoms*). It is formulated as follows:

$$\mathbf{x} = \sum_{t=1}^{T} \mathbf{a}_t^{(\mathrm{X})} \cdot h_t = \mathbf{A}^{(\mathrm{X})} \cdot \mathbf{h}, \quad (5)$$

where $\mathbf{x} \in \mathcal{R}^{p \times 1}$ represents the spectrum of one frame, T is the total number of speech atoms, $\mathbf{A}^{(\mathrm{X})} = [\mathbf{a}_1^{(\mathrm{X})}, \mathbf{a}_2^{(\mathrm{X})}, \cdots, \mathbf{a}_T^{(\mathrm{X})}] \in \mathcal{R}^{p \times T}$ is the dictionary of speech atoms built from training source speech, $\mathbf{a}_t^{(\mathrm{X})}$ is the t^{th} speech atom which has the same dimension as $\mathbf{x}$, $\mathbf{h} = [h_1, h_2, \cdots, h_T] \in \mathcal{R}^{T \times 1}$ is the non-negative weight or activation vector and h_t is the activation of the t^{th} speech atom.

Therefore, the spectrogram of each source utterance can be represented as:

$$\mathbf{X} = \mathbf{A}^{(\mathrm{X})} \cdot \mathbf{H}, \quad (6)$$

where $\mathbf{X} \in \mathcal{R}^{p \times M}$ is the source spectrogram, and $\mathbf{H} \in \mathcal{R}^{T \times M}$ is the activation matrix, the column vector of which is the activation vector in Eq. (5).

In order to generate converted speech spectrogram, we assume that the aligned source and target dictionaries share the same activation matrix. To this end, we represent the converted spectrogram as:

$$\mathbf{Y} = \mathbf{A}^{(\mathrm{Y})} \cdot \mathbf{H}, \quad (7)$$

where $\mathbf{Y} \in \mathcal{R}^{q \times M}$ is the converted spectrogram, and $\mathbf{A}^{(\mathrm{Y})} \in \mathcal{R}^{q \times T}$ is the dictionary of the target speech atoms from target training data.

The illustration of Eq. (6) and (7) is presented in Fig. 1. The source and target dictionaries $\mathbf{A}^{(\mathrm{X})}$ and $\mathbf{A}^{(\mathrm{Y})}$ are constructed from parallel training data and they remain the same during the conversion phase. During the conversion phase, the source spectrogram is given and the activation matrix is obtained as a solution of non-negative matrix factorization as in [18]. Then, the activation matrix estimated from Eq. (6) is then directly employed in Eq. (7) to generate the converted spectrogram.

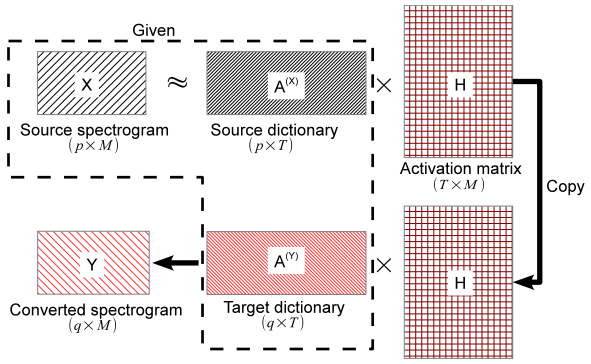


Fig. 1. Illustration of non-negative spectrogram factorization for exemplar-based voice conversion

3.2. Non-negative spectrogram deconvolution (NMD)

Although temporal constraints can be included in the estimation of activation matrix by using multiple-frame exemplars as source speech atoms, the converted speech spectrogram is still generated frame-by-frame. In order to utilize temporal context in the generation process of the converted spectrogram, we propose *non-negative spectrogram deconvolution (NMD)* method for exemplar-based voice conversion. In the NMD method, a spectrogram is represented as a convolution of exemplars and activations. The idea is formulated as follows:

$$\mathbf{X} = \sum_{l=1}^{L} \mathbf{A}_l^{(\mathrm{X})} \cdot \overset{\rightarrow(l-1)}{\mathbf{H}}, \tag{8}$$

$$\mathbf{Y} = \sum_{l=1}^{L} \mathbf{A}_l^{(\mathrm{Y})} \cdot \overset{\rightarrow(l-1)}{\mathbf{H}}, \tag{9}$$

where $\mathbf{A}_l^{(\mathrm{X})} \in \mathcal{R}^{p\times T}$ and $\mathbf{A}_l^{(\mathrm{Y})} \in \mathcal{R}^{q\times T}$ are the matrices consisting of the l^{th} frame of the source and target atoms, respectively, L is the number of adjacent frames within an examplar and $\mathbf{H}$ is the activation (weights) matrix as that in Eq. (6). $\overset{\rightarrow(l-1)}{(\cdot)}$ operator shifts the matrix entries (columns) to the right by $(l-1)$ units. In practice, several consecutive frames of an exact frame can be stacked into one supervector to represent the exact frame for constructing the source dictionary $\mathbf{A}_l^{(\mathrm{X})} \in \mathcal{R}^{p\times T}$. Therefore, $p = L \times d$ other than $p = d$, where d is the dimension the spectrum. During conversion, a source spectrogram $\mathbf{X}$ is first decomposed to estimate the activation matrix, and then the converted speech spectrogram $\mathbf{Y}$ is generated as a convolution of the target speech atoms and the corresponding activation matrix. The activation matrix is obtained by minimizing the generalized Kullback-Leibler divergence as explained in [19].

3.3. Dictionary construction

As discussed above, dictionary is important for both estimating the activation matrix and generating the converted speech signal. Before introducing how to construct dictionary, we first introduce the related features used to represent spectrum. In this work, the STRAIGHT [23] system is employed to extract spectral envelope and fundamental frequency (F0). The following three features are involved in this study:

a) *Magnitude spectra (MSP)*: Magnitude spectra consist a sequence of spectral envelopes extracted by STRAIGHT. We use 513 dimensional spectra. Magnitude spectra can be passed to STRAIGHT for reconstructing speech signal. In this work, target dictionary and converted spectrogram are always represented by MSP.

b) *Mel-scale magnitude spectra (MMSP)*: MMSP is obtained by passing the magnitude spectrogram to a 23-channel Mel-scale filter-bank. The minimum frequency is set to be 133.33 Hz, and the maximum frequency is set to be 6,855.5 Hz. In this work, MMSP is only used in the source dictionary to estimate the activation matrix but not for synthesizing speech.

c) *Mel-cepstral coefficient* (MCC): MCC is obtained by employing mel-cepstral analysis technique on the magnitude spectrogram and keeping 24 coefficients as the feature. During synthesis, MCCs are converted back to magnitude spectrogram, which is then passed to the STRAIGHT synthesis filter to reconstruct speech signal. In this work, MCC is only used in the JD-GMM method and in the dynamic time warping to align two parallel utterances.

Given one pair of parallel utterances from source and target, the following process is employed to construct the dictionary.

1) Extract magnitude spectrogram (spectral envelopes) from both source and target speech signal using STRAIGHT;

2) Apply mel-cepstral analysis [24] on the spectrograms to obtain mel-cepstral coefficients (MCCs);

3) Apply 23-channel Mel-scale filter-bank to obtain 23-dimensional MMSP;

4) Perform dynamic time warping on the source and target MCC sequence to align the speech to obtain source-target frame pairs;

5) Apply the alignment information to the source and target spectrograms. The resulting spectrum pairs are stored in the source and target dictionaries (column vectors), respectively.

The above five steps are applied for all the parallel training utterances. All the spectrum pairs (column vectors in source and target dictionaries) are used as speech atoms. In order to include multiple frames, consecutive frames are stacked into a super-vector to represent one frame. We note that for simple explanation, same features (both spectral envelopes) are used in step 5. As the size of the activation matrix is independent of the dimensionality of the features (column dimensionality), therefore, 23-dimensional MMSP can be used to replace 513-dimensional MSP in the source dictionary. While *513-dimensional MSP is always used in the target dictionary for synthesizing speech purpose*. More details will be discussed in Section 4.

4. EXPERIMENTS

To evaluate the proposed methods, we conduct experiments using the VOICES database [25]. Male-to-female and female-to-male conversions are conducted. For each conversion, 10 utterances from each speaker are selected as training data and 20 utterances, which are not included in the training data, are used as testing data.

In the experiments, three methods are compared. They are summarized as follows:

a) *JD-GMM*: The joint density Gaussian mixture model method (Section 2). The number of Gaussian components is set to be 32.

b) *NMF*: The proposed non-negative spectrogram factorization method (Section 3.1).

c) *NMD*: The proposed non-negative spectrogram deconvolution method (Section 3.2).

In the JD-GMM method, 24-dimensional MCC features are used to represent spectral envelope and to synthesize speech signal, while in NMF and NMD method, 513-dimensional MSP is used in the target dictionary and to synthesize speech signal. Log-scale F0 is converted by equalizing the mean and variance of the source and target speech.

4.1. Objective evaluation

Two objective measures are employed to evaluate the proposed method objectively. The first objective measure is spectral distortion: *mel-cepstral distortion* (MCD), which is calculated between a converted frame and the corresponding original target frame. We note that the frame alignment is obtained by performing dynamic time warping between parallel source and target sentences. The MCD for the m^{th} frame is calculated as:

$$\text{MCD[dB]} = \frac{10}{\log 10}\sqrt{2\sum_{d=1}^{24}(c_{m,d} - c_{m,d}^{\text{conv}})}, \qquad (10)$$

where, M is frame number in one utterance, $c_{m,d}$ and $c_{m,d}^{\text{conv}}$ are the d^{th} dimension of the original target and converted MCCs of the m^{th} frame, respectively. We report the average MCD value over all the frames. A lower MCD value indicates smaller distortion. The second objective measure is the *correlation coefficient*, which is calculated between the original target and the converted MCC parameter trajectories dimension-by-dimension. The correlation coefficient γ_d of the d^{th} MCC trajectory is computed as follows:

$$\gamma_d = \frac{\sum_{m=1}^{M}(c_{m,d} - \overline{c_d})(c_{m,d}^{\text{conv}} - \overline{c_d^{\text{conv}}})}{\sqrt{\sum_{m=1}^{M}(c_{m,d} - \overline{c_d})^2}\sqrt{\sum_{m=1}^{M}(c_{m,d}^{\text{conv}} - \overline{c_d^{\text{conv}}})^2}}, \qquad (11)$$

where $\overline{c_d}$ and $\overline{c_d^{\text{conv}}}$ are the mean values of the original target and converted MCCs of the d^{th} dimension, respectively. We note that correlation coefficient is calculated sentence-by-sentence and we report the average correlation coefficient. Different from MCD, correlation coefficient focuses on the trajectory-level similarity, which is not affected by the mean and variance of the MCC trajectory, and has been used to measure the fundamental frequency trajectory similarity [5, 26]. Bigger correlation coefficient indicates Higher similarity between the original target and the converted MCC trajectories. We report the average correlation coefficient over all dimension.

In order to obtain comparable MCD and correlation results, in the NMF and NMD method, mel-cepstral analysis is applied to the converted spectrogram to get the 24-dimensional MCCs for computing MCD and correlation coefficient. Both MCD and correlation coefficient results reported in this work are averaged over the conversion pairs.

As shown in Eq. (6) and (7), and Fig. 1, the dimensionality of the activation matrix is independent of the dimensionality of the exemplars in both the source and the target dictionaries. Therefore, we first evaluate the performance of NMF using different features in source dictionary for estimating the activation matrix. *We note that for all the experiments, target dictionary always use the 513-dimensional magnitude spectra*, as the target dictionary does not affect the activation matrix and also is used to synthesize speech signal.

As discussed above, the dimensionality of the spectral envelope from STRAIGHT is 513 (1024-point FFT). If the original magnitude spectra (MSP) are used to estimate the activation matrix, as illustrated in Fig. 1, the dimensionality of the source dictionary $\mathbf{A}^{(\text{X})}$ will be $513 \times T$, assuming that each exemplar spans only one frame. If each exemplar spans 11 frame, the dimensionality of the source dictionary $\mathbf{A}^{(\text{X})}$ will be $5,643 \times T$, where T is the number of atoms. The huge dimensionality of the source dictionary will increase the computation and memory usage considerably. To reduce computation and memory usage, low-dimensional features will be a better choice. In this study, we propose to use 23-dimensional MMSP instead of the 513-dimensional original MSP to make the source dictionary for estimating the activation matrix. While the target dictionary reminds same as discussed above.

Table 1 presents the spectral distortions and correlations of NMF using 513-dimensional MSP and 23-dimensional MMSP in the source dictionary. Here, an exemplar spans only one frame. The results show that, even the dimensionality is reduced from 513 to 23, the distortion only increases 0.06 dB, and the correlation decreases by 0.003. The benefit of using 23-dimensional MMSP instead of 513-dimensional MSP in source dictionary to represent speech signal is that more consecutive frames can be included in the exemplar to estimate the activation matrix without increasing the computation cost and memory usage too much.

Table 1. Comparison of NMF results using 513-dimensional magnitude spectra (MSP) and 23-dimensional Mel-scale magnitude spectra (MMSP) in the source dictionary $\mathbf{A}^{(\text{X})}$. 513-dimensional MSP is always used in the target dictionary $\mathbf{A}^{(\text{Y})}$.

Features in source dictionary $\mathbf{A}^{(\text{X})}$	MCD (dB)	Correlation
MSP (513 dimensions)	5.47	0.439
MMSP (23 dimensions)	5.53	0.436

We then evaluate the performance of NMF using multiple frames in an exemplar for source dictionary. The spectral distortion results as a function of the window size (number of consecutive frames) of an exemplar is presented in Fig. 2. For Mel-scale magnitude spectra, the window size of exemplar is varied. While for 513-dimensional magnitude spectra, only one frame spectrum is employed in the exemplar due to computation restrictions discussed above. The results show that when the window size is larger than 3, 23-dimensional MMSP yields lower MCD and higher correlation coefficient than 513-dimensional MSP. NMF with exemplar using MMSP and spanning 9 frames gives the lowest spectral distortion. We note that the dimensionality of exemplar using MMSP and spanning 9 frames is $23 \times 9 = 207$, which is still much smaller than 513. The correlation results in Fig. 3 agree well with the spectral distortion results.

Next, we evaluate the proposed non-negative deconvolution (NMD) method using 23-dimensional MMSP. As shown above, 9

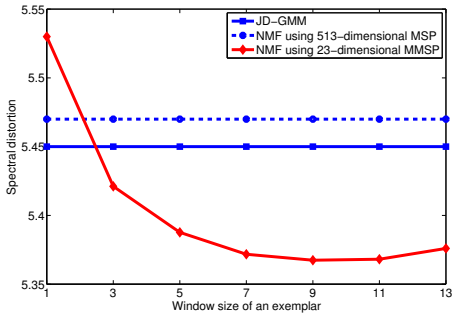


Fig. 2. The spectral distortion results of NMF method using different features with the baseline JD-GMM method as a reference

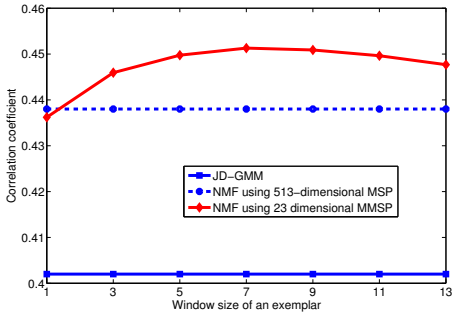


Fig. 3. The correlation coefficient results of NMF method using different features with the baseline JD-GMM method as a reference

frame exemplars give lowest distortion in the NMF method, therefore, in the NMD method, we stack 9 consecutive frames of an exact frame to represent the exact frame. Therefore, In Eq. (1), $p = 9 \times 23 = 207$. The spectral distortion results are presented in Fig. 4, as a function of the window size of an exemplar. Comparing with JD-GMM method, we observe that NMD method always obtains lower spectral distortion. NMD and NMF methods have similar performance in terms of spectral distortion when the window size is 5 or 7. The correlation coefficient results are shown in Fig. 5. It clearly shows that NMD has the highest correlation coefficients in all the cases. We note that different from NMF, NMD method utilizes multiples target frames (an exemplar) not only to estimate the activation matrix but also to generate the converted spectrogram.

4.2. Subjective evaluation

To assess the similarity of the converted speech to the target speech, a similarity preference listening test was conducted. The JD-GMM, and the two proposed methods: NMF and NMD are compared. 10 converted utterances from each method were randomly selected, including 5 utterances from the male-to-female conversion and the other 5 utterances from the female-to-male conversion. 11 subjects were asked to listen to a reference target speech and then the three converted speech samples representing the three methods. After that they were asked to decide which speech sample is more closer to the reference target speech sample. The preference scores with 95% confidence interval are presented in Fig. 6. We can clearly observe that the proposed NMF and NMD methods are both able to generate speech samples which are more similar to the target speaker than the

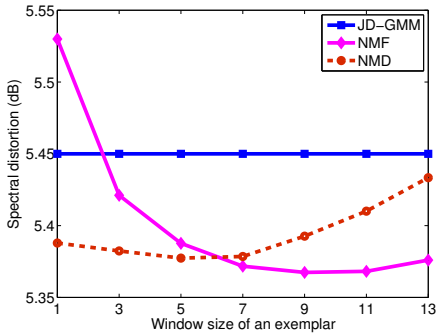


Fig. 4. Comparison of the spectral distortion results of JD-GMM, NMF and NMD methods as a function of the window size of an exemplar

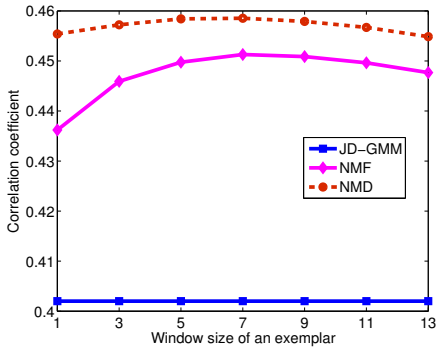


Fig. 5. Comparison of the correlation results of JD-GMM, NMF and NMD methods as a function of the window size of an exemplar

baseline JD-GMM method. We note that during the listening test, when the subjects are not able to distinguish the similarity across speech samples, they prefer to choose the one which gives better quality. Therefore, the similarity can reflect the speech quality of the three methods to some degree.

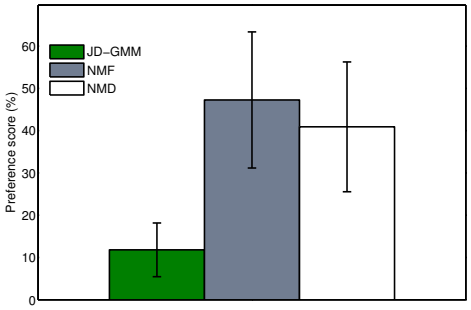


Fig. 6. Similarity results of the preference score with 95% confidence interval

5. CONCLUSIONS

In this paper, we proposed an exemplar-based voice conversion method utilizing the matrix/spectrogram factorization techniques. Two implementations, non-negative spectrogram factorization and non-negative spectrogram deconvolution, are proposed to use original target spectrogram directly without any dimension reduction to synthesize the converted speech. The experiment results show the proposed method outperforms the conventional joint density Gaussian mixture model considerably.

6. ACKNOWLEDGEMENT

The work of Tuomas Virtanen (projects no. 258708) and Tomi Kinnunen was supported by Academy of Finland (projects no. 253120). The authors would like to thank all the listeners who take part in the subjective evaluation test.

7. REFERENCES

[1] T. Toda, A.W. Black, and K. Tokuda, "Voice conversion based on maximum-likelihood estimation of spectral parameter trajectory," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 8, pp. 2222–2235, 2007.

[2] D. Saito, S. Watanabe, A. Nakamura, and N. Minematsu, "Statistical voice conversion based on noisy channel model," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 6, pp. 1784–1794, 2012.

[3] M. Narendranath, H.A. Murthy, S. Rajendran, and B. Yegnanarayana, "Transformation of formants for voice conversion using artificial neural networks," *Speech communication*, vol. 16, no. 2, pp. 207–216, 1995.

[4] B. Gillet and S. King, "Transforming F0 contours," in *Proceedings of Eurospeech*, 2003, pp. 101–104.

[5] Z.Z. Wu, T. Kinnunen, E.S. Chng, and H. Li, "Text-independent F0 transformation with non-parallel data for voice conversion," in *Eleventh Annual Conference of the International Speech Communication Association*, 2010.

[6] C. H. Wu, C. C. Hsia, T. H. Liu, and J. F. Wang, "Voice conversion using duration-embedded bi-HMMs for expressive speech synthesis," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 4, pp. 1109–1116, 2006.

[7] M. Abe, S. Nakamura, K. Shikano, and H. Kuwabara, "Voice conversion through vector quantization," in *ICASSP 1998*.

[8] A. Kain and M.W. Macon, "Spectral voice conversion for text-to-speech synthesis," in *ICASSP 1998*.

[9] Y. Stylianou, O. Cappé, and E. Moulines, "Continuous probabilistic transform for voice conversion," *IEEE Transactions on Speech and Audio Processing*, vol. 6, no. 2, pp. 131–142, 1998.

[10] E. Helander, T. Virtanen, J. Nurminen, and M. Gabbouj, "Voice conversion using partial least squares regression," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 5, pp. 912–921, 2010.

[11] H. Zen, Y. Nankaku, and K. Tokuda, "Continuous stochastic feature mapping based on trajectory hmms," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 2, pp. 417–430, 2011.

[12] Z. Wu, T. Kinnunen, E. Chng, and H. Li, "Mixture of factor analyzers using priors from non-parallel speech for voice conversion," *Signal Processing Letters, IEEE*, 2012.

[13] V. Popa, H. Silen, J. Nurminen, and M. Gabbouj, "Local linear transformation for voice conversion," in *ICASSP 2012*.

[14] S. Desai, E.V. Raghavendra, .B Yegnanarayana, A.W. Black, and K. Prahallad, "Voice conversion using artificial neural networks," in *ICASSP 2009*.

[15] P. Song, Y.Q. Bao, L. Zhao, and C.R. Zou, "Voice conversion using support vector regression," *Electronics letters*, vol. 47, no. 18, pp. 1045–1046, 2011.

[16] E. Helander, H. Silén, T. Virtanen, and M. Gabbouj, "Voice conversion using dynamic kernel partial least squares regression," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 3, pp. 806–817, 2012.

[17] Z. Wu, E.S. Chng, and H. Li, "Conditional restricted boltzmann machine for voice conversion," in *the first IEEE China Summit and International Conference on Signal and Information Processing (ChinaSIP)*, 2013.

[18] J.F. Gemmeke, T. Virtanen, and A. Hurmalainen, "Exemplar-based sparse representations for noise robust automatic speech recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 7, pp. 2067–2080, 2011.

[19] A. Hurmalainen, J. Gemmeke, and T. Virtanen, "Non-negative matrix deconvolution in noise robust speech recognition," in *ICASSP 2011*.

[20] T.N. Sainath, B. Ramabhadran, D. Nahamoo, D. Kanevsky, D. Van Compernolle, K. Demuynck, J.F. Gemmeke, J.R. Bellegarda, and S. Sundaram, "Exemplar-based processing for speech recognition: An overview," *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 98 –113, nov. 2012.

[21] R. Takashima, T. Takiguchi, and Y. Ariki, "Exemplar-based voice conversion in noisy environment," in *Spoken Language Technology Workshop (SLT), 2012 IEEE*. IEEE, 2012, pp. 313–317.

[22] Y. Chen, M. Chu, E. Chang, J. Liu, and R. Liu, "Voice conversion with smoothed GMM and MAP adaptation," in *Eurospeech-2003*.

[23] H. Kawahara, I. Masuda-Katsuse, and A. de Cheveigné, "Restructuring speech representations using a pitch-adaptive time–frequency smoothing and an instantaneous-frequency-based f0 extraction: Possible role of a repetitive structure in sounds," *Speech communication*, vol. 27, no. 3, pp. 187–207, 1999.

[24] T. Fukada, K. Tokuda, T. Kobayashi, and S. Imai, "An adaptive algorithm for mel-cepstral analysis of speech," in *ICASSP 1992*.

[25] A. KAIN, "High resolution voice transformation," *Ph. D. Thesis, OGI School of Science and Engineering, Oregon Health and Science University*, 2001.

[26] Y. Qian, Z. Wu, B. Gao, and F.K. Soong, "Improved prosody generation by maximizing joint probability of state and longer units," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 6, pp. 1702–1710, 2011.

Mage - Reactive articulatory feature control of HMM-based parametric speech synthesis

Maria Astrinaki[1], *Alexis Moinet*[1], *Junichi Yamagishi*[2,3], *Korin Richmond* [2], *Zhen-Hua Ling*[4], *Simon King*[2], *Thierry Dutoit*[1]

[1]Circuit Theory and Signal Processing Lab, Numediart Institute, University of Mons, Belgium
[2]The Centre for Speech Technology Research, University of Edinburgh, Edinburgh, UK
[3]National Institute of Informatics, Tokyo, Japan
[4]University of Science and Technology of China (USTC), China

maria.astrinaki@umons.ac.be, alexis.moinet@umons.ac.be, jyamagis@inf.ed.ac.uk
korin@cstr.ed.ac.uk, zhling@ustc.edu, simon.king@ed.ac.uk, thierry.dutoit@umons.ac.be

Abstract

In this paper, we present the integration of articulatory control into MAGE, a framework for realtime and interactive (reactive) parametric speech synthesis using hidden Markov models (HMMs). MAGE is based on the speech synthesis engine from HTS and uses acoustic features (spectrum and f_0) to model and synthesize speech. In this work, we replace the standard acoustic models with models combining acoustic and articulatory features, such as tongue, lips and jaw positions. We then use feature-space-switched articulatory-to-acoustic regression matrices to enable us to control the spectral acoustic features by manipulating the articulatory features. Combining this synthesis model with MAGE allows us to interactively and intuitively modify phones synthesized in real time, for example transforming one phone into another, by controlling the configuration of the articulators in a visual display.

Index Terms: speech synthesis, reactive, articulators

1. Introduction

For human beings, speech is one of the richest and most sophisticated modalities used for communication. It involves complex production and perception mechanisms and it varies in quality. It is a highly reactive and interactive process, with complex timing, involving the all the articulators (tongue, lips, jaw, lungs, etc.), even the hands. Artificially synthesized speech has been explored for decades and several methods have been developed, such as formant synthesis [1], diphone synthesis [2], articulatory speech synthesis [3], unit selection synthesis [4] and statistical parametric synthesis [5] resulting in various Text-To-Speech (TTS) systems. Nowadays, TTS systems are very intelligible and natural, and can be expressive, but they are also static, they do not support user interaction and they are not sensitive to environmental conditions. Although there has been great progress in terms of intelligibility and naturalness, there is still place for improvement when considering interaction. This "deaf" design of conventional TTS systems is limiting the involvement of the user and the vocal expression influenced by its environment, leaving no room for expression and creativity.

In recent years, there has been an emerging interest in applications that need reactivity and expressivity in speech production. There are different ways of approaching the idea of working beyond TTS. One approach to move further than the standard TTS paradigm comes with MAGE [6], one of the first methods proposed for reactive HMM-based speech and singing synthesis. It is a modified version of the HMM-based parametric speech synthesis approach that has become a mainstream speech synthesis method [7], [8]. MAGE allows reactive control of prosody, context, speaking style and speech quality. It is able to synthesize highly intelligible and smooth speech, it is flexible, and supports adaptation and interpolation methods, combined with a very small footprint and computational weight; advantages inherited from the original system, HTS [9]. However, the quality of the output is constrained by the quality of the training data and the user controls. It is still difficult, even for trained users to produce meaningful expressivity due to the complexity of the speech itself and to the abstract representation of speech through statistical models.

The controls that have been available, have been over the acoustic features, as used in the conventional HMM-based speech synthesis, which are parameters required by a vocoder. Such parameters do not necessarily have a "physical" or "intuitive" meaning to the user. However, the physical nature of human speech production means that an articulatory parameterization of speech has interesting properties. The articulatory features describe the quantitive positions and the continuous movements of a group of human articulators, such as tongue, jaws, lips, velum. Such features have relatively slow and consistent evolution through time, they are not influenced in the same way by acoustic noise and other environmental conditions. They can provide a straightforward and simple explanation for speech characteristics and they provide meaningful interpretation of the speech production to the user.

A method for integrating articulatory features in HMM-based speech synthesis has already been proposed [10], [11], where the articulatory features were recorded using electromagnetic articulography (EMA). In this method, a unified acoustic-articulatory model is trained and a piecewise linear transform is adopted to model the dependency of the acoustic features on the articulatory features. During synthesis, the articulatory features are generated from the previously trained models. Then, these generated articulatory features can be manipulated in arbitrary ways which in turn affect the generation of acoustic features. In this way, the characteristics of the synthetic speech can be controlled via an articulatory representation. The motivation of the work described here is to see wether reactive articulatory control is possible, to evaluate the results and then to explore the potential and the possibilities of different user applications

(see Section 2.2) that take advantage of the physical nature and stability of the articulators.

The paper is organized as follows. Section 2 gives a brief overview of the reactive HMM-based speech synthesis approach called MAGE. Section 3 describes our proposed method in detail. Section 4 describes the articulatory control application we developed as a proof of concept as well as the challenges faced for the evaluation and testing. Section 5 presents the future targets and Section 6 gives the conclusions of this work.

2. Reactive HMM-based speech and singing synthesis

MAGE is based on the HMM-based parametric speech synthesis method, which it extends in order to support realtime architecture and multithreaded control. As it is based on HTS, it inherits its features, advantages and drawbacks [5]. The contribution of MAGE is that it opens the enclosed processing loop of the conventional system and allows reactive user control over the available contextual information, the speech prosody and speaking style and quality. Moreover, it provides a simple C++ API, allowing reactive HMM-based speech synthesis to be easily integrated into reactive and realtime frameworks [12], [13]; run in various devices and create different prototypes [14], [15].

2.1. Overview of MAGE

One important feature of MAGE is that it uses multiple threads, and each thread can be affected by the user which allows accurate and precise control over the different production levels of the artificial speech. As illustrated in Figure 1, MAGE integrates three main threads: the *label thread*, the *parameter generation thread* and the *audio generation thread*. Three queues are shared between threads: the *label queue*, the *parameter queue* and the *sample queue*.

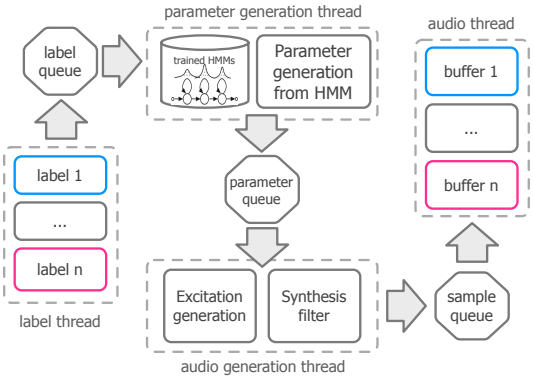


Figure 1: MAGE: reactive parameter generation using multiple threads and shared queues.

Briefly, the *label thread* controls the input sequence of the phonetic labels, by pushing the received phonetic labels onto the *label queue*. Then, the *parameter generation thread* reads from the *label queue* one phonetic label at a time. For that single label the speech parameters are generated (sequences of spectral and excitation parameters including first and second derivatives of the static features), which are locally-maximized using only the current phonetic label and, if available, the two previous labels. In other words, for every single input phonetic label, the feature vectors are estimated by taking into account the HMMs of these specific labels and the input user controls. The generated speech parameters are stored in the *parameter queue*. As shown in [16], the impact of the local maximization of the generated parameters is very small. Then, the *audio generation thread* will generate the actual speech samples and store them in the *sample queue* so that the system's *audio thread* will access them and deliver them to the user. Further details of the MAGE reactive parameter estimation can be found in [17].

Accessing and controlling every thread has a different impact over the synthesized speech. The *label thread* can provide contextual phoneme control, the *parameter generation thread* can reactively modify the way the available models are used for the parameter sequence generation [17], and finally the *audio generation thread* manipulates reactively the vocoding of every sample, resulting in prosody and voice quality controls. The delay in applying any received user control varies between a single speech sample and a phonetic label depending on the thread that is being accessed.

2.2. Potential applications

MAGE aims to combine simple prototyping with meaningful gestural control, through an appropriate mapping and to bring synthetic speech to performative use cases. It can enable a broad set of new types of design and architectures for speech synthesis applications, such as silent speech communication, entertainment and gaming, assistive applications for speech impaired people and performing arts.

Several real life and scientific applications target the creation and use of unique personalized voices, with certain speech characteristics. For example, in the field of new interfaces for musical expression and performing arts, it would be possible to create a voice that has a very specific speaking style that could also gradually change during the performance or adapt to the feedback of the audience. Regarding avatars and gaming applications, users would be able to select, customize and refine the voice used by their avatar. It could also be used in a GPS, where the accent of the used voice changes depending on the position, as the user is driving through a country. The same principle applies to movie dubbing applications or assistive communication devices for speech impaired people, where a voice can be adapted and personalized according to the past and the personality of every person or character.

We believe that by adding articulatory control to MAGE not only will the range of the potential applications be enriched, but also it will be possible to achieve a deeper understanding of articulatory speech production mechanisms. Applications targeting the fields of speech pedagogy, linguistics and speech therapy in particular will be able to help people come to understand how certain phones are produced at the articulatory level, providing instant acoustic feedback. Our attempt to implement this is discussed in the following sections.

3. Reactive articulatory feature control

In this work, MAGE is modified in order to generate and alter articulatory features. Given the unified acoustic-articulatory model and a set of phonetic labels, it is possible to reactively generate the target speech samples. Simultaneously, it is possible to influence the generated articulatory features by replacing them with the user input. In this way, we can achieve the goal of altering the generated speech samples at the articulatory level rather than directly at the acoustic level.

3.1. Feature-Space-Switched Multiple Regression HMMs

In HMM-based speech synthesis, a sequence of contextual phonetic labels is used to predict an optimal state sequence and the duration, in frames, of each state. In the case of acoustic features (i.e. spectral parameters), a state j corresponds to a multivariate Gaussian distribution whose parameters are $\boldsymbol{\mu}_j$, its mean vector, and $\boldsymbol{\Sigma}_j$, its covariance matrix. Given these parameters, the sequence of states and their durations are used to generate an optimal sequence of acoustic features $\boldsymbol{X}$ that are then combined with synthetic source parameters to synthesize speech using, for instance, a vocoder. Note that in MAGE, as presented in Section 2, the state sequence and the computation of parameters are performed locally, label by label, as opposed to the one-pass approach that we have just described.

This framework for HMM-based parametric speech synthesis has been expanded in [11] so that the acoustic models become dependent on articulatory features. One of the methods presented is called "feature-space-switched multiple regression HMM" (FSS-MRHMM). MRHMM consists in replacing the mean vector $\boldsymbol{\mu}_j$ of each state by a linear combination of synthetic articulatory features $\boldsymbol{\xi}_t$ and $\boldsymbol{\mu}_j$, before computing the optimal sequence of acoustic features. Therefore, the Gaussian distribution for each frame t is defined as

$$b_j(\boldsymbol{x}_t|\boldsymbol{y}_t) = \mathcal{N}(\boldsymbol{x}_t; \boldsymbol{A}_t\boldsymbol{\xi}_t + \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j) \quad (1)$$

with $\boldsymbol{x}_t$ a vector of static acoustic features and their first and second derivatives. $\boldsymbol{A}_t$ is the articulatory-to-acoustic projection matrix and $\boldsymbol{\xi}_t$ is an expanded articulatory feature vector, which means it contains $[\boldsymbol{y}_t^\intercal, 1]$, with $\boldsymbol{y}_t$ a vector of static articulatory features and their first and second derivatives. Normally, $\boldsymbol{y}_t$ is generated using standard HMM-based synthesis with its own specific models of articulatory features. However, as explained in Section 3.2, it can also be replaced by other values, thus modifying the identity of the synthetic phones.

In the particular case of FSS-MRHMM, a finite set of M matrices $\{\boldsymbol{A}_1, \ldots, \boldsymbol{A}_M\}$ is trained along with an M-mixture Gaussian mixture model (GMM) of the articulatory space. Then, at synthesis time, instead of a single Gaussian the probability density function of each state is written as

$$b_j(\boldsymbol{x}_t|\boldsymbol{y}_t) = \sum_{k=1}^{M} \zeta_k(t)\mathcal{N}(\boldsymbol{x}_t; \boldsymbol{A}_k\boldsymbol{\xi}_t + \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j) \quad (2)$$

where $\zeta_k(t)$ is the probability for the mixture component k given $\boldsymbol{y}_t$. However, with such a model, the parameter generation would require to use an EM-based iterative estimation. This cannot be applied in the context of a reactive application such as MAGE and we simplified it by considering only the mixture with maximum $\zeta_k(t)$, as proposed in [11]. Therefore, Equation 2 is rewritten as

$$k_t = \underset{k=[1,\ldots,M]}{\operatorname{argmax}}\ \zeta_k(t) \quad (3)$$

$$b_j(\boldsymbol{x}_t|\boldsymbol{y}_t) = \mathcal{N}(\boldsymbol{x}_t; \boldsymbol{A}_{k_t}\boldsymbol{\xi}_t + \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j) \quad (4)$$

Further details on the training of the different models (acoustic, articulatory, GMM and $\boldsymbol{A}_k$) can be found in [11].

3.2. Reactive synthesis using articulatory features

During synthesis, a given sequence of context-dependent phonetic labels is used to concatenate the context-dependent HMMs. The articulatory and acoustic features are then predicted from the sentence HMMs by means of a maximum output probability parameter generation algorithm that incorporates dynamic features. It is possible though during synthesis that $\boldsymbol{\xi}_t$, the generated articulatory features, may be modified or replaced either by user input or according to phonetic knowledge (as explained in [11]). Hence, the corresponding acoustic features are regenerated, using Equations 3 and 4, in order to reflect those articulatory changes. The speech waveform is then synthesized from the generated mel-cepstral and f_0 parameter sequences using Mel Log Spectrum Approximation (MLSA) filter [18], with pulse-train (voiced frames) or white-noise excitation (unvoiced frames).

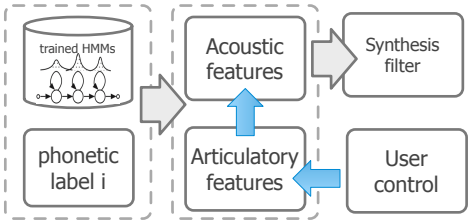


Figure 2: Generation of acoustic features with articulatory control using acoustic-articulatory model and user input controls.

As explained in Section 2, only the current phone label and, if available, the two previous labels are taken into account for the feature generation and therefore the generated parameter trajectories are locally-maximized. Here, for every phone label, the articulatory features are generated, taking into account the control of the user over the articulators (if any). Then, the acoustic features are generated in order to correspond to these articulatory features, as illustrated in Figure 2. Finally, the speech waveform is synthesized. Note that the feature generation is taking place in the *parameter generation thread*, and therefore the application of user control has a delay of one phone label.

The aim of this approach is to use the articulatory features in order to replace to some extent the predefined context and to modify the acoustic features accordingly. In other words, the intention is to reactively alter a given context and its acoustic features by using only modifications over the articulatory features provided by the user.

4. Reactive articulatory control application

To evaluate the proposed method requires an implementation that combines a graphical user interface (GUI) with the MAGE synthesizer. Such an application[1] is essential for multiple reasons. First and foremost, it allows us to assess the quality of the final speech samples. But, moreover we can explore how this output is influenced by fast changing articulatory inputs as well as how proficient users must be at controlling such features.

4.1. Graphical user interface design

The design of the graphical user interface was highly dependent on the database we used for the reactive synthesis. In this work for our experiments and reactive synthesis we have used a multichannel articulatory database containing the acoustic waveform

[1] A video demonstration of the presented system can be found in `https://vimeo.com/67404386`.

recorded concurrently with EMA data of a male British English speaker. Six EMA receivers were used, and for each receiver three dimensional coordinates were recorded as described in [10]. However, only two dimensions were used in the experiments here (front-to-back and bottom-to-top) resulting in a total of 12 static articulatory features.

Based on these six EMA points with two dimensional movements, we designed the GUI illustrated in Figure 3. The GUI depicts a two dimensional midsagittal view of the vocal tract drawn using 124 points. The six EMA points are represented as white circles placed on the articulators as described in [10] (indicated by the red arrows). The position of these EMA points can be reactively controlled by the user using a mouse or touch screen. There are no limits to the possible position of the EMA points providing to the user 12 degrees of freedom. This means that the user is free to place these points in coordinates that are "unnatural" either from a physical point of view or as sequence of movements.

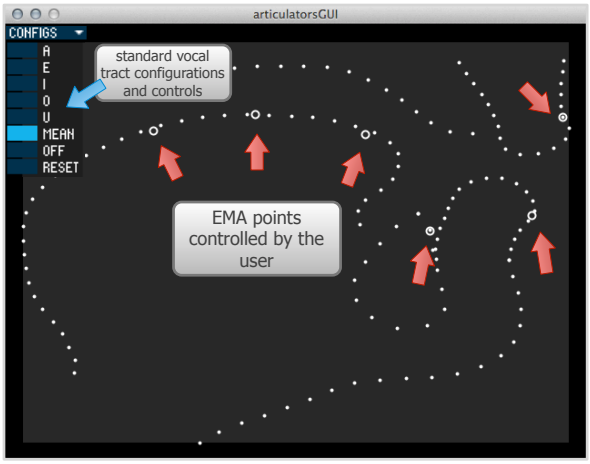


Figure 3: Instance of the graphical user interface showing the available configurations and controls, the six EMA points (red arrows) that can be reactively moved and predefined configurations of vocal tracts that can be applied (blue arrow).

On the left of the interface there is a menu listing predefined vocal tract shapes and EMA sets. These predefined configurations of vocal tract shapes were obtained from speaker-dependent magnetic resonance imaging (MRI) scans and electromagnetic articulography (EMA) data, as described in [19]. The GUI shows vocal tract shapes from models trained on MRI scans combined with EMA points translated in the MRI coordinate space. When the user selects one of these vocal tract and EMA configurations, his previous controls are instantly overwritten and the selected shape is displayed. The user by selecting one EMA point from the GUI is able to move it in the two dimensional MRI coordinate space. The current shape of the vocal tract it is not reactively altered so that the user will have a reference point to the initial configuration chosen. However, it is possible to transform interactively the shape of the vocal tract given the user controls by using specific transformation matrixes. Note here that the controls of the user take place in the MRI space (GUI) and not in the EMA space (synthesis).This means that the controls of the user have to be appropriately transformed in order to be an acceptable input for MAGE. This task is controlled by the interface.

The interface also allows to reset the synthesis by clicking on the "reset" button. It is also possible to stop using the reactive articulatory controls and use the generated articulatory data by clicking on the "off" button.

4.2. Synthesis

The final part of the application, generating the speech waveform, is implemented by the MAGE reactive speech synthesizer. The graphical user interface sends to MAGE the modifications applied by the user through open sound control (OSC) messages [20].When received, these modifications over the EMA points are taken into account to generate the corresponding articulatory features. These features are used to estimate the acoustic features, then will give the final speech samples with only one phonetic label delay, as explained in Section 3. Let us note here that the articulatory features provided by the user overwrite the estimated articulatory features, $\boldsymbol{\xi}_t$ used in Equation 4. Therefore, the first and second derivatives used are the ones from the contextually estimated articulatory features.

4.3. Challenges

One of the aims of this work is to allow the user to reactively control the speaking style as well as the content by modifying the articulatory features. However, the movement of the articulators is so fast that the user is not able to input the expected movements fast enough through the interface. Therefore, instead of trying to contextually manipulate a full phrase we decided to try and transform only one vowel into another vowel. This simplifies the problem of the fast changing articulatory features but introduces the problem of the duration of the synthesized vowel. If it is synthesized with the standard model duration is too short to be intelligible. Hence, in this case we synthesize long vowels by increasing the generated duration of every state, using a bigger scale factor for the stable state (i.e. 10, 20 or 50), followed by a long pause.

The EMA points can be placed at any position and with any speed. This results in movements of the articulators that do not always respect the "physiology" and "mechanisms" of the human articulators. In other words, these input movements have not been "seen" during the training phase and, consequently, the models cannot accurately estimate them. Hence, when these "new" articulatory features will be used to generate the acoustic features it is highly probable that they will give an "unstable" result. In order to tackle the problem of the extended contextual control and to minimize the possible instabilities caused by the extreme movements of the articulators, MAGE constrains the contextual control only over the voiced frames.

As a test case, the user is asked to listen to a vowel synthesized using phonetic labels and by moving the six EMA points reactively from the interface he transforms it into another target vowel. However, after some exploratory tests we see that this approach requires from the user to move the six EMA points, in the two dimensional MRI space (12 degrees of freedom) accurately enough so that he will achieve the acoustic target. Such a task is very demanding and rather difficult, and in most cases the user does not reach the target. What makes this task more difficult is that we use context-dependent model. Although the FSS-MRHMM approach can determine the regression matrices without using context information, the $\boldsymbol{\mu}_j$ and $\boldsymbol{\Sigma}_j$ as shown in Equation 4 are still context-dependent. More specifically, the required modifications over the articulatory features (EMA points) in order to acoustically achieve a target, differ depending on the initial phone synthesized using the provided label.

A solution to this would be to use "tailored" context features, where the vowel identities are removed from the question set for acoustic model clustering and therefore better articulatory controllability can be achieved [11].

5. Future work

Based on the preliminary testing of the application we see some essential modifications regarding the interface. It is important to either decrease the degrees of freedom available to the user or to allow the manipulation of only some EMA points while the system provides the correct coordinates for the remaining ones. For example, a case would be where the user is allowed to manipulate only the three EMA points over the tongue. while the remaining three are automatically adjusted. Providing a "color-coded map" denoting the "accepted" or "suggested" regions of every EMA point could advice or guide the user to choose suitable coordinate sets. This will help the user to have a better understanding of the required modifications of the articulators in order to achieve the acoustically desired target. However, such a simplification of the interface must by all means be combined with using "tailored" context feature during synthesis for better articulatory controllability.

Based on such a framework, it would be meaningful to conduct user studies and listening tests. The user studies will show us how users manipulate the acoustic space by means of articulatory control as well as how skilled a user should be. The listening tests will help us to measure how other listeners perceive the result of these manipulations. Initially, as explained above, the user is asked to transform a given vowel to a target vowel only by controlling the articulatory features. The success of the user will be determined by objectively evaluating the acoustic and articulatory features generated by taking into account or not the user input for the target vowel. The same test can be conducted by using monosyllabic words embedded into a carrier sentence in order to conduct a second vowel identity modification experiment. Then, it would be interesting to see how other users perceive these modifications, and in addition to the objective evaluation, we would like to perform also some listening tests to subjectively evaluate performance on the vowel modification task.

6. Conclusions

In this paper we have presented a method that enables reactive articulatory control over HMM-based parametric speech synthesis using the MAGE framework. We present also an application that enables the user to reactively control the position of the articulators through a graphical user interface. We see that reactive articulatory control is feasible, and combined with an interface allows us to explore different aspects of the speech production. However, we realize that the manipulation of the articulators by the user, even though it seems rather straightforward, is very demanding and difficult. It is very easy to transform a phone into another random phone while experimenting with the interface, but it becomes rather complicated when a specific target vowel modification is asked. There are aspects of the system that would benefit from improvement. Currently, there are no restrictions over the user manipulation patterns, but probably limiting or "guiding" the possible user controls might lead in more distinguishable vowel modifications. Subjective and objective evaluations are essential in order to assess the final quality of the system. By conducting user studies we want to evaluate the efficiency of the user to achieve certain vowel or monosyllabic word targets. Through listening tests we want also to evaluate how the output of these reactive modifications over the articulatory features is perceived acoustically.

7. References

[1] R. Carlson and B. Granstrom, "A text-to-speech system based entirely on rules," in *Proc. of ICASSP*, 1976.

[2] F. Charpentier and M. Stella, "Diphone synthesis using an overlap-add technique for speech waveforms concatenation," in *Proc. of ICASSP*, vol. 11, 1986, pp. 2015–2018.

[3] B. Brent and W. J. Strong, "Windbag – a vocal-tract analog speech synthesizer," *Acoustical Society of America*, vol. 45, 309(A), 1969.

[4] A. Hunt and A. Black, "Unit selection in a concatenative speech synthesis system using a large speech database," in *Proc. of ICASSP*, 1996, pp. 373–376.

[5] H. Zen, K. Tokuda, and A. W. Black, "Statistical parametric speech synthesis," *Speech Communication*, vol. 51, pp. 1039–1064, 2009.

[6] M. Astrinaki, A. Moinet, G. Wilfart, N. d´Alessandro, and T. Dutoit. (2010, September) Mage platform for performative speech synthesis. [Online]. Available: http://mage.numediart.org/

[7] T. Yoshimura, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, "Simultaneous modeling of spectrum, pitch and duration in HMM-based speech synthesis," *Proc. Eurospeech*, vol. 83, no. 11, pp. 2347–2350, 1999.

[8] K. Tokuda, H. Zen, and A. Black, "HMM-based approach to multilingual speech synthesis," *Text to speech synthesis: New paradigms and advances*, pp. 135–153, 2004.

[9] K. Oura. (2010, Sept) HMM-based speech synthesis system (hts). [Online]. Available: http://hts.sp.nitech.ac.jp/

[10] Z.-H. Ling, K. Richmond, J. Yamagishi, and R.-H. Wang, "Integrating articulatory features into HMM-based parametric speech synthesis," *IEEE Transactions On Audio Speech And Language Processing*, vol. 17, no. 6, pp. 1171–1185, 2009.

[11] Z. Ling, K. Richmond, and J. Yamagishi, "Articulatory control of HMM-based parametric speech synthesis using feature-space-switched multiple regression," *IEEE TASLP*, vol. 21, no. 1, pp. 207–219, 2013.

[12] M. Puckette. (2009, September) Pure data. [Online]. Available: http://puredata.info/

[13] Z. Lieberman, T. Watson, A. Castro, and etc. (2009, September) openframeworks. [Online]. Available: http://www.openframeworks.cc

[14] R. A. Clark, M. A. Konkiewicz, M. Astrinak, and J. Yamagishi, "Reactive control of expressive speech synthesis using kinect skeleton tracking," Tech. Rep. 30, December 2012.

[15] M. Astrinaki, A. Moinet, N. d´Alessandro, and T. Dutoit, "Pure data external for reactive HMM-based speech and singing synthesis," *Proc. of the 16th Int. Conference on Digital Audio Effects (DAFx-13)*, 2013.

[16] M. Astrinaki, N. d'Alessandro, B. Picart, T. Drugman, and T. Dutoit, "Reactive and continuous control of HMM-based speech synthesis," in *Spoken Language Technology Workshop (SLT), 2012 IEEE*, 2012, pp. 252–257.

[17] M. Astrinaki, N. d´Alessandro, L. Reboursière, A. Moinet, and T. Dutoit, "MAGE 2.00: New features and its application in the development of a talking guitar," in *Proc. of the 13th Conference on New Interfaces for Musical Expression (NIME´13)*, 2013.

[18] K. Sumita and R. Members, "Mel log spectrum approximation (mlsa) filter for speech synthesis," *Electronics and Communications in Japan*, vol. 6, no. 2, pp. 10–18, 1983.

[19] K. Richmond and S. Renals, "Ultrax: An animated midsagittal vocal tract display for speech therapy," in *Proc. Interspeech*, 2012.

[20] A. Schmeder, A. Freed, and D. Wessel. (2009, September) opensoundcontrol.org. [Online]. Available: http://opensoundcontrol.org

Systematic Database Creation for Expressive Singing Voice Synthesis Control

Martí Umbert, Jordi Bonada, Merlijn Blaauw

Music Technology Group, Universitat Pompeu Fabra, Barcelona, Spain

marti.umbert@upf.edu, jordi.bonada@upf.edu, merlijn.blaauw@upf.edu

Abstract

In the context of singing voice synthesis, the generation of the synthesizer controls is a key aspect to obtain expressive performances. In our case, we use a system that selects, transforms and concatenates units of short melodic contours from a recorded database. This paper proposes a systematic procedure for the creation of such database. The aim is to cover relevant style-dependent combinations of features such as note duration, pitch interval and note strength. The higher the percentage of covered combinations is, the less transformed the units will be in order to match a target score. At the same time, it is also important that units are musically meaningful according to the target style. In order to create a style-dependent database, the melodic combinations of features to cover are identified, statistically modeled and grouped by similarity. Then, short melodic exercises of four measures are created following a dynamic programming algorithm. The Viterbi cost functions deal with the statistically observed context transitions, harmony, position within the measure and readability. The final systematic score database is formed by the sequence of the obtained melodic exercises.

Index Terms: expressive singing voice synthesis, unit selection, database creation

1. Introduction

Expressive performances have attracted the interest of researchers for the last years. Providing expression and emotions has been a goal for many types of synthesizers, from instruments to speech and singing voice synthesis. An important issue is to provide control data to the synthesizer which represent a given emotion, expression or style. Several strategies have been proposed to cover the target expressive space.

In the case of the reconstructive phrase modeling system [1], the Synful orchestra achieves musical expressivity by deriving parameters from database of real songs according to a target song. In their work, working with a systematic database is discarded because it would reduce the expressive power of the system since the amount of expressive articulations is too large. In the current work we propose a possible solution to overcome this problem with a double criteria: to cover the frequent articulations with a musical criteria at the same time.

Other approaches have designed scores by taking musical phrases from real repertoires, as in [2] for a violin synthesizer. A first manual step is done to select a repertoire that covers the target features and that is musically relevant. Then, an algorithm is used to select and transform trajectories to cover different note transition features (e.g. articulations, note intervals) and intro note features (accents, duration and dynamics).

In concatenative speech synthesis, articulations are captured from real recordings of sounds, which span from single or groups of phonemes (diphones, triphones), to words or sentences. In [3], unit selection of the recorded sounds has been studied. However, when preparing scripts to achieve phoneme coverage, other aspects can be taken into account, such as how difficult words are to read or grammatical correctness of a formed sentence, which has been addressed in [4] and also in [5].

In emotional speech synthesis, statistical modeling techniques have been proposed to model speaking styles as in [6]. In this case, similar to recording real musical scores, no specific constraints are given to the recording scripts. Emotion related scripts are recorded and then modeled with HMMs.

Statistical modeling of singing style has also been used in [7] with focus to relative pitch, vibrato and dynamics using context dependent HMMs. In this case, real recordings are used and therefore no previous study is performed with respect to which scripts given the target style.

This paper studies the generation of a set of exercises that represent to some extent a given style properties and melodies. These exercises will then be recorded by one singer in one style. Pitch, dynamics and note durations will be extracted and used within a unit selection-based approach to generate expressive controls of a singing voice synthesis system.

In our case we did not choose the option of generating melodic exercises directly from real repertoires. These typically have the disadvantage of being redundant, so only a portion of an entire song introduces new note sequences. Also, in order to select which part of a song to include as an exercise, it should be carefully studied. Therefore, we obtain melodic exercises by concatenating short melodic units generated in a systematic way, also including musical knowledge as explained in section 2. First, a set of scores are statistically analyzed in order to know which feature values (note strengths and figures and pitch intervals in semitones) should be covered, their relevance and how these are connected. Then, dynamic programming is applied in order to generate melodic exercises as sequences of concatenated units. Finally, in section 3 conclusions and future work are presented.

2. Database creation

2.1. Units versus contexts

The basic elements of our systematic process of melodic exercises creation are units made up as sequences from one to three notes surrounded by a previous and following note or silence. An example is shown in Figure 1. In this paper a note is defined mainly by the following properties: note strength, note duration (seconds), and the figure and pitch interval with the next one. Note strength (NS) is a measure for the accentuation of a note beat within a bar. Figure interval (FI) refers to the relationship between two consecutive note durations and the same applies to pitch interval (PI) with respect to the note frequencies. This data is shown in Figure 2.

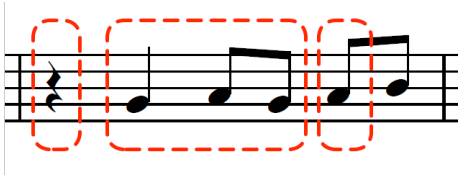


Figure 1: *Unit of three notes with preceding silence and following note.*

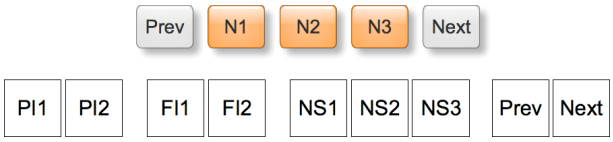


Figure 2: *Unit and context features.*

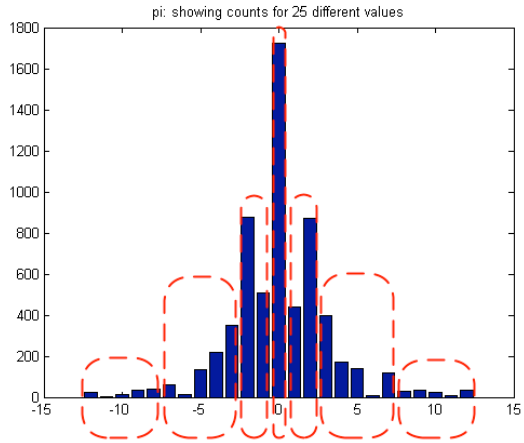


Figure 3: *Pitch interval distribution (in semitones) and clusters.*

For each note property there are many possible combinations, which imply a great amount of units, specially in the case of sequences of three notes. This relates to the goal of the systematic database, which is to cover a high amount of relevant note combinations. Therefore, the coverage criteria is not defined with respect to the units but related to a higher abstract unit or context. Each context comprises several possible units.

Thus, the relationship between units and contexts has to be defined by grouping the set of values of each note property into clusters. Once the clusters are set, it is possible to statistically analyze the transition probabilities between contexts. Both steps are explained in the following section.

2.2. Statistical analysis and clustering

In order to study the set of note properties that need to be covered, a set of songs belonging to the same style have been processed using Music21 [8], a Python toolkit to process music in symbolic form.

Since most of the processed units are three notes long, and each note is defined in terms of its strength, duration, and figure and pitch intervals, the possible number of units is enormous. As explained in the previous section, to reduce the amount of units to cover, these are clustered into similar contexts.

In general, clusters have been organized so that close values are represented by the same cluster. In the case of pitch interval clusters, it has been taken into account that within the same cluster all pitch intervals correspond to only ascending or descending intervals, since we do not want to transform an ascending pitch contour to synthesize a descending one (and vice versa). Therefore, an interval of zero semitones (same consecutive notes) is grouped in a separate cluster. In the case of the figure interval, clusters do not need to follow the same constraint concerning the direction of the interval (ascending or descending). Note strength clusters have been grouped according to the note accentuation within a measure.

In Figure 3 and Table 1, the values distribution for the figure interval and their clustering is shown. The same information is presented with respect to note strength in Figure 4 and Table 2, and concerning pitch interval in Figure 5 and Table 3.

Using this cluster representation, the context frequencies have been counted and the 90% most common ones have been selected to be covered, generating a list of 993 contexts of three notes. Also, the amount of connections between these selected contexts (by overlapping two or one notes or just concatenating them) has been computed to measure the transition probabilities among contexts. These contexts are a higher level representation of 1480 units.

Table 1: *Pitch interval cluster values.*

Cluster	Range of values
1	[-12, -8]
2	[-7, -3]
3	[-2, -1]
4	[0]
5	[1, 2]
6	[3, 7]
7	[8, 12]

2.3. Melodic exercises generation

The Viterbi algorithm has been used in order to generate the sequence of melodic exercises of the systematic database. The temporal resolution, or tick, of each melodic exercise is defined by the minimum note length. In our case we have used a tick of an eighth note. The sequence of ticks defines a note strength grid which is used in order to know which units fit at each position in time. The note strength grid generation is explained in section 2.3.1.

At each (forward) step of the Viterbi algorithm, the accumulated cost of inserting a given database unit at a certain tick is computed using a set of cost functions explained in section 2.3.2. These cost functions handle the transitions between units according to the statistical information at context level computed as explained in section 2.2. The cost functions also measure whether an instance fits in the grid and reusing a context is penalized. Harmony is managed by the preset accompaniment chords (which convey the target style) of the melodic exercises and how these and the unit notes match. Inserting silences in the middle of the exercise is also favored considering readability, in order to help the singer to breath in the middle of the performance. Also, the generated note pitches are constrained to the singers tessitura in order to facilitate singing the exercises.

The following subsections explain the process followed to generate the melodic exercises as sequence of three note long units. In a similar way exercises of two and one notes were

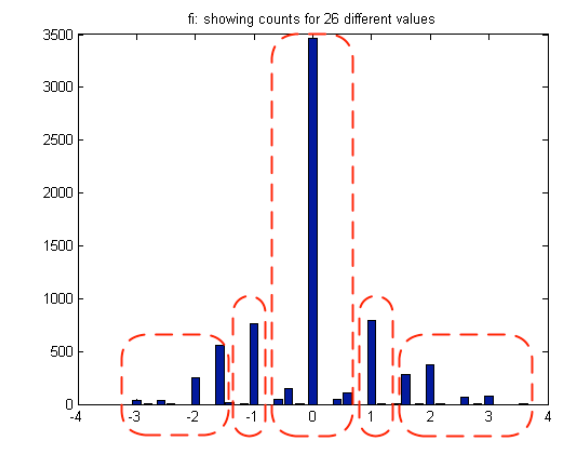


Figure 4: *Figure interval distribution (in octaves) and clusters.*

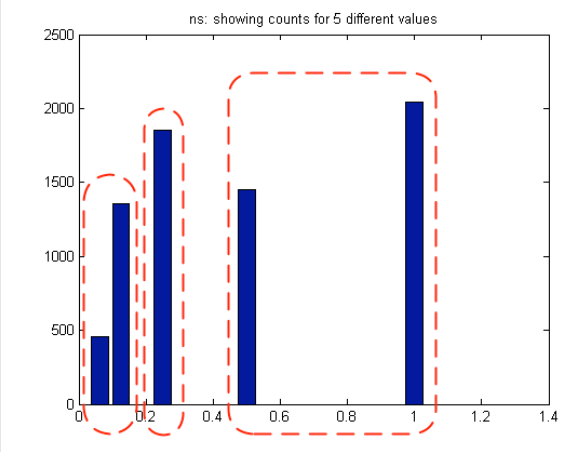


Figure 5: *Note strength distribution and clusters.*

generated. In these cases, the previous and following notes are considered to be silences, so the Viterbi algorithm was not necessary since unit overlapping does not apply. These exercises were generated in a more straightforward manner by taking one value per cluster to generate the contexts to cover.

Table 2: *Figure interval cluster values.*

Cluster	Range of values
1	[-3, -1.585]
2	[-1.41, -1]
3	[-0.585, 0.585]
4	[1, 1.415]
5	[1.585, 3.585]

2.3.1. *Note strength grid*

Given the minimum note length that will be used in the systematic score, a grid can be generated which sets where notes can be placed and which their note strengths are at those positions. The length of this grid is related to the amount of measures per exercise.

For a minimum note length of an eighth note, the note strength grid for a single measure (4 beats, 8 ticks) is musically defined as shown in the following vector:

$$[1, 0.125, 0.25, 0.125, 0.5, 0.125, 0.25, 0.125] \quad (1)$$

2.3.2. *Cost measures*

The accumulated cost for an evaluated node of the Viterbi matrix is evaluated by several cost measures.

The first computed cost checks whether the note strengths features of the unit match the note strengths related to the tick position where is is intended to be inserted. If the unit does not fit, then it is not necessary to check all the other costs, and the total cost is set to infinity. For units that do fit, the cost is set to zero.

The second computed cost relates to the transition between units. The result of the statistical analysis provides this cost for an overlapping of two, one or zero notes (concatenation). These transition is computed for the current selected unit with respect to all possible previous units.

Table 3: *Note strength cluster values.*

Cluster	Range of values
1	[0.5, 1]
2	[0.25]
3	[0.125, 0.625]

Since the aim is to have the highest coverage possible with the minimum amount of melodic exercises, context repetition is taken account for penalization. Therefore, a history of all previously selected contexts is kept, so that if in the currently evaluated node path there is a context repetition, a cost proportional to the amount of repetitions is added. Although some context repetitions may appear in the final score, this cost favors the selection different contexts.

The harmony cost takes into account the chords for the melodic exercises. The same sequence of chords has been predefined for all exercises in order to make it easy for the singer: C7 (1st bar), Am7 (2nd bar), Dm (3rd bar 1st half), G7 (3rd bar 2nd half), C7 (4th bar). Those notes with cost zero are the ones belonging to the chord. Otherwise, it is more costly to add notes which do not match with the chord note information. In Table 4 the harmony costs are shown relating which notes are favored (zero cost) per chord and which ones are more penalized (non-zero cost).

Finally, since melodic exercises are four measures long (plus one as a break between exercises), and in order to make them more easy to sing, a silence has been included in the middle, at the end of the second measure and at the beginning of the third one. Several tick candidates for inserting the pause are considered in the Viterbi paths and the least costly one is chosen.

Table 4: *Harmony costs.*

Bar	Chord	C	D	E	F	G	A	B
1	C7	0	1	0	2	0	1	0
2	Am7	0	1	0	2	0	0	1
3	Dm	1	0	1	0	2	0	2
3	G7	2	0	2	0	0	1	0
4	C7	0	1	0	2	0	1	0.5

2.3.3. Stop criteria

The algorithm stops generating melodic exercises depending on two conditions. The first one is related to the coverage. If all 993 contexts have been selected (one unit per context is enough) after the generation of a melodic exercise, the generation of exercises is stopped. This is controlled by the history of selected contexts as explained in the previous section.

The second stop criteria is related to the available recording session duration and the tempo of the generated score. If the accumulated duration of all exercises reaches the recording time, given the amount of measures per exercise and the bpm, then no more melodic exercises are generated.

2.3.4. Results

The systematic script has been generated by taking 57 jazz standard songs, setting the tessitura to one octave, a tempo of 71 bpm and a limit for the recording time of one hour. These constraints generate a recording script of 236 exercises and a coverage of 82% of contexts.

The generated melodic exercises as concatenation of three note long units can be downloaded in pdf and audio files are online at: `http://www.dtic.upf.edu/~mumbert/ssw8/`.

3. Conclusions

A system for the systematic generation of melodic exercises has been presented. The aim of such melodic exercises is to cover the statistically and musically more relevant note combinations in terms of note strength and figure and pitch intervals. The concepts of units up to three notes and their feature clustering in order to group them into high level contexts has been presented in order to define the coverage criteria.

We plan to perform an evaluation in order to prove that the generated systematic score is representative of the songs statistically analyzed. This can be proven by taking a set of target songs different from the analyzed set but belonging to the same style. Our unit selection-based approach can be used in order to retrieve units from the systematic database and to measure the degree of transformation (note duration, pitch shifting) that these require to match the target. Also, the difference of note strengths between the selected units and the target units can be computed. Finally, phrasing aspects are important. For example, the length of retrieved sequences of consecutive units from the database is a measure of representativeness. The longer these sequences are, the longer the recorded contours used by our framework will be.

These measures should differ from generating a systematic score from another style database and computing the levels of unit transformations and consecutive unit sequences lengths.

We also plan to improve the grouping process of note properties into clusters. This could also be done following a K-means algorithm. Also, the central values within a cluster should be more represented in the final score than extreme values.

Also, once the systematic score is recorded, and the expressive contours extracted, the complete framework with both symbolic and expressive trajectories will be tested to generate the expressive contours.

4. References

[1] E. Lindemann, "Music synthesis with reconstructive phrase modeling," *Signal Processing Magazine, IEEE*, vol. 24, no. 2, pp. 80–91, 2007.

[2] A. Pérez, "Enhancing spectral synthesis techniques with performance gestures using the violin as a case study," Ph.D. dissertation, Universitat Pompeu Fabra, 2009.

[3] A. J. Hunt and A. W. Black, "Unit selection in a concatenative speech synthesis system using a large speech database," in *Acoustics, Speech, and Signal Processing, 1996. ICASSP-96. Conference Proceedings., 1996 IEEE International Conference on*, vol. 1. IEEE, 1996, pp. 373–376.

[4] M. Dong, L. Cen, P. Chan, and H. Li, "Readability consideration in speech synthesis recording script selection."

[5] S. Fitt, "Using real words for recording diphones," 2001.

[6] M. Tachibana, J. Yamagishi, T. Masuko, and T. Kobayashi, "Speech synthesis with various emotional expressions and speaking styles by style interpolation and morphing," *IEICE - Trans. Inf. Syst.*, vol. E88-D, no. 11, pp. 2484–2491, Nov. 2005.

[7] K. Saino, M. Tachibana, and H. Kenmochi, "A singing style modeling system for singing voice synthesizers." in *INTERSPEECH*. ISCA, 2010, pp. 2894–2897.

[8] M. S. Cuthbert and C. Ariza, "music21: A toolkit for computer-aided musicology and symbolic music data," in *Proceedings of the International Symposium on Music Information Retrieval*, vol. 11, 2010, pp. 637–42.

EXPRESSIVE SPEECH SYNTHESIS: SYNTHESISING AMBIGUITY

Matthew P. Aylett[1 2], *Blaise Potard*[2], *Christopher J. Pidcock*[2]

University of Edinburgh, Informatics[1]
CereProc Ltd.[2]
Edinburgh, UK
matthewa@inf.ed.ac.uk

ABSTRACT

Previous work in HCI has shown that ambiguity, normally avoided in interaction design, can contribute to a user's engagement by increasing interest and uncertainty. In this work, we create and evaluate synthetic utterances where there is a conflict between text content, and the emotion in the voice. We show that: 1) text content measurably alters the negative/positive perception of a spoken utterance, 2) changes in voice quality also produce this effect, 3) when the voice quality and text content are conflicting the result is a synthesised ambiguous utterance. Results were analysed using an evaluation/activation space. Whereas the effect of text content was restricted to the negative/positive dimension (valence), voice quality also had a significant effect on how active or passive the utterance was perceived (activation).

Index Terms: speech synthesis, unit selection, expressive speech synthesis, emotion, prosody.

1. INTRODUCTION AND STATEMENT OF RELATION WITH PRIOR WORK

In many systems, speech synthesis is required purely to communicate neutral dynamic information to a user, for example their bank balance or the time of an appointment. However, as computer applications become more complex, for example by simulating environments, or taking on the role of a trainer or tutor, the interaction required with users also becomes more complex. In such systems, user engagement becomes more important, and in order to build systems which can create a compelling sense of engagement, Human Computer Interaction (HCI) research has begun to look at alternatives to the dominant approach of user-centered design. Two alternatives to the traditional HCI approach are ludic design, which focuses on the importance of encouraging playfulness in a design[1], and experience-centred design, which focuses on the sense of experience that a system would like to engender in a user[2]. In these design approaches, ambiguity, normally avoided in interface design, can be harnessed to encourage intrigue, mystery and delight[3]. Speech synthesis is a key enabling technology for pervasive design, and in order to face the new challenges of affective, eyes-free and mobile systems, speech synthesis technology needs to offer designers the flexibility and functionality that can support these new design methodologies. This presents a challenge for speech synthesis, both in terms of creating ambiguity in synthetic utterances, and in evaluating this ambiguity.

Ambiguity is often the result of a tension between opposing perceptions. It is this tension which can add to a user's curiosity and engagement. This is quite different from neutrality, where there are no dominant or contrasting perceptions. For example 'hot and cold' is ambiguous, whereas 'warm' is neutral. In natural speech, ambiguity is often used to create a specific effect, for example irony. One definition of irony is *an expression or utterance marked by a deliberate contrast between apparent and intended meaning.[4]* One method used by human speakers to generate irony, is to use a contrasting emotion to the content spoken, for example "What a brilliant day" said with an angry or stressed voice. Contrasting meaning and emotion in this way creates a complex picture of the speaker. It conveys more than the straightforward utterance "What a horrible day", because the tension between the voice and the content suggest a complex internal state which in turn adds to the sense of character.

Current speech synthesis systems typically produce neutral speech, although more recently, work in expressive speech synthesis has examined how to create speech which *unambiguously* conveys an emotion or an underlying expressive goal. This work has a long tradition of focusing on evaluating a distinct set of between three and nine, extreme, sometimes termed *primitive* emotional states, such as disgust, fear, anger, joy, sadness, and surprise [5]. This presents a problem for creating ambiguous utterances, because a very strong emotion in the voice will dominate the perception of the utterance. Instead a more controlled approach is required which can offset other features in the utterance. The CereVoice speech synthesis system uses a distinct set of sub-corpora containing different *voice qualities* to achieve a more subtle change in the perceived emotion in an utterance.

Voice quality is an important factor in the perception of emotion in speech[6]. However, unlike speech rate and pitch, which can be modified relatively easily using digital signal processing techniques such as PSOLA, modifying voice quality is more difficult, especially if it is important to retain naturalness. Rather than modifying speech to create the effect, an alternative approach is to record different voice qualities and use them directly during concatenative synthesis. This approach has been applied to diphone synthesis [7] and has been extended to unit selection in the CereVoice system which uses pre-recorded voice quality sub-corpora in unit selection [8]. This is different from other unit selection approaches which have instead examined the use of sub-corpora of specific emotions, e.g. [9] where Happy, Angry and Neutral sub-corpora were incorporated into an emotional voice in Festival. By focusing on voice quality rather than specific emotions, CereVoice allows a combination of DSP techniques and unit selection to craft a more varied and subtle set of speech styles[10].

As with [7] three styles of voice quality (VQ) are available: Neutral (the default for the recorded corpora) and two sub-corpora of lax (calm) and stressed (tense) voice quality. Adding an XML tag in the speech biases the selection of the units to come from the sub-corpora. However, the extent to which this unit-selection VQ approach suc-

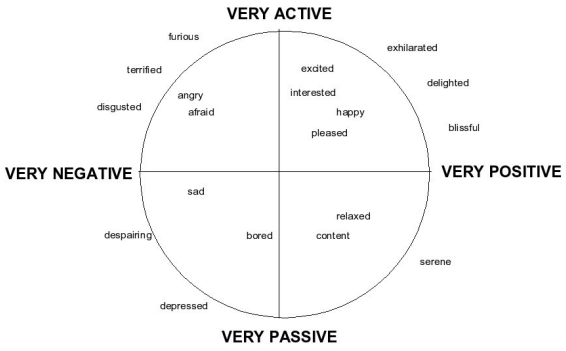


Fig. 1. *Activation/Evaluation Space*

ceeds in conveying a negative/positive perception of the utterance has not been formally evaluated until now.

In order to both evaluate this approach, as well as evaluate the success or failure of creating an ambiguous utterance, we require an evaluation methodology which allows a response to be shifted depending on competing factors. In this work, we adapt the approach taken by FEELTRACE[11] and evaluate utterances within the *activation/evaluation space*.

FEELTRACE was developed specifically for assessing gradual changes in emotion by allowing subjects to place the emotion in a two dimensional space called the evaluation/activation space. This space is based on previous work in psychology [12, 13] and regards emotions as having two components, a valence which varies from negative to positive, and an activation which varies from passive to active (See Figure 1). In this way, rather than asking subjects which emotion they perceive in an utterance, the subject chooses a point in this two dimensional space. This approach is especially powerful for detecting shifts in emotion.

2. RESEARCH QUESTIONS

In order to create a conflict between voice quality and text content, sentences were chosen with content intended to be both negative and positive. Neutral sentences, and natural speech with neutral, lax and stressed voice qualities, were used as controls.

Our research questions were:

RQ1: Does voice quality change equate to a change in the positive/negative (valence) perception of an utterance?

RQ2: If so, can we use a mismatch between voice quality and text content to create ambiguity in synthetic utterances?

3. METHODOLOGY

Voice quality is one feature among many that effect the perception of emotion in speech. As we wished to discover the effect of *voice quality change* only, we used the same approach as Hofer et al [9], and asked subjects to rate the emotion in the synthetic and natural speech by choosing a position in the activation/evaluation space (Figure 1). We also asked subjects to rate naturalness on a 5 point scale (Bad/Poor/Fair/Good/Excellent). The experiment was carried out online (see Figure 2) using 14 English native speakers. Subjects were requested to use headphones. There were three factors in the experiment: Synthesis/natural speech **SYN**, Stressed/Lax/Neutral voice quality **VQ**,

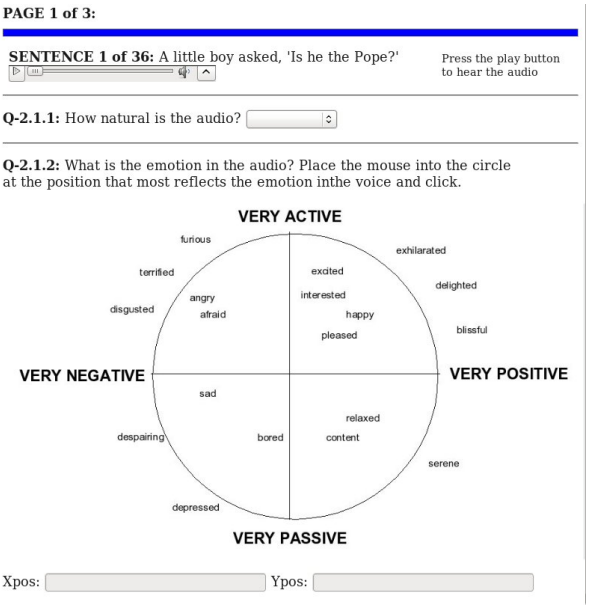


Fig. 2. Online Experimental Setup

and Positive/Negative/Neutral Text content **TCONT**. Although sentences were present for all conditions in synthetic speech, natural speech sentences were missing for: Positive Text with Stressed Voice Quality, Negative Text with Lax Voice Quality and Positive/Negative Text with Neutral Voice Quality. Synthesis was generated using the CereProc Sarah RP female voice with natural stimuli held out from the speech database.

Positive and Negative text content was selected from online news materials by evaluating sentences using the dictionary of affect[14]. The dictionary of affect gives valence and activation scores to 8742 emotional words. Positive sentences were selected which contained more positive words and the converse for negative sentences. All sentences were manually checked in order to ensure the overall semantic meaning also matched the desired text category. There were 12 sentences in each category.

Our hypotheses were:

1. H1: There is a significant difference in perceived valence between utterances with stressed and lax voice qualities, for both natural and synthetic speech.
2. H2: Text content has a significant influence on perceived valence.
3. H3: Where text content mismatches voice quality, the perceived valence moves towards the neutral point in the valence scale due to the opposing perceptions creating by the ambiguity.

4. RESULTS

4.1. Naturalness

It is important to assess naturalness when testing speech synthesis in any context as very poor naturalness will confound other perception results and call into question the utility of any process that reduces naturalness below an acceptable level.

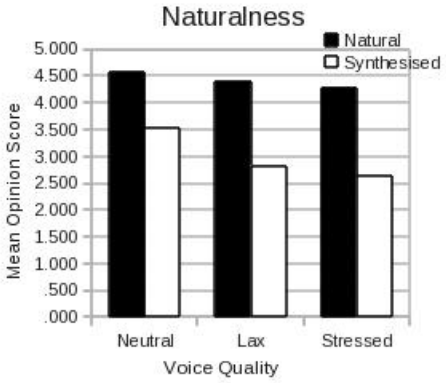


Fig. 3. Mean opinion score by voice quality and by natural and synthetic speech.

Figure 3 shows naturalness expressed as a mean opinion score (MOS) grouped by voice quality. A grouped univariate ANOVA analysis with two factors, **SYN** and **VQ** was carried out. Both factors were significant (SYN: $F(1, 462)=201.98$, $p<0.001$), (VQ: $F(2, 462)=11.00$, $p<0.001$) with a just significant between factor interaction ($F(2, 462)=3.13$, $p<0.05$). Post-hoc Tukey tests showed a significant drop in naturalness for stressed and lax synthesised sentences compared to Neutral sentences suggesting that concatenating mixed sub-corpora causes an increase in synthetic artifacts. However this drop in quality is typically less than 0.5 MOS. The naturalness of neutral utterances compares favourably with state of the art systems (typically around 3.5)[15]. Results using MOS scores should be treated with care as there is a strong argument that the underlying subject data should not be treated as parametric data. However MOS is a default standard in speech synthesis and using MOS allows a multifactor analysis of the data using a grouped ANOVA analysis. Although MOS data is rarely Gaussian, an ANOVA analysis is acceptable based on the sampling theorem providing each cell has sufficient data points (commonly 10 or above).

4.2. Voice Quality

Figure 4a shows the mean values for neutral sentences with different voice quality within the activation/evaluation space by **SYN** and **VQ**. The means of natural utterances are shown in black, synthetic utterances in grey.

A grouped multivariate ANOVA analysis was significant for voice quality, for both valence ($F(2, 186)=16.75$, $p<0.001$) and activation ($F(2, 186)=57.48$, $p<0.001$), with a significant interaction between **VQ** and **SYN** for activation ($F(2,186)=6.03$, $p<0.005$). Post-hoc tests showed that, in general, natural sentences with lax and stressed voice quality were rated further from the centre of the activation/evaluation space than synthesised sentences. However synthesised sentences showed similar, if less marked, effects of voice quality than natural sentences.

Although one aim of voice quality change is to modify the valence, there is also a strong effect on the perception of activation. Lax voice quality is associated with low activation and positive valence, and stressed voice quality is associated with high activation and negative valence.

In addition, we must note that our neutral voice quality was rated very positively. For commercial systems, voice talents are chosen for having pleasant positive voices and this can undermine the use of a neutral voice quality in such voices as a representative control.

However, results show that altering voice quality affects the perception of valence and activation in an utterance. Although the effect for valence is significant only between stressed and other voice qualities this allows to accept hypothesis H1.

4.3. Effect of Text Content

Due to missing cells for text content in the natural stimuli set, a second ANOVA was carried out on synthetic materials only. A grouped multivariate analysis was carried out with **VQ** and **TCONT** factors. **VQ** results supported those for the neutral sentences ($F(2,325)=18.15$, $p<0.001$ for valence and $F(2,325)=53.25$, $p<0.001$ for activation). **TCONT** had no significant effect on naturalness or activation but showed a significant effect for valence ($F(2,157)=10.43$, $p<0.001$). Post-hoc Tukey tests showed a significant difference between negative text content and neutral and positive content ($p<0.05$) but not between neutral and positive text content. Means for text content for neutral voice quality utterances only are shown in Figure 4b.

Results show that text content affects the perception of valence in a synthesised utterance. This allows us to accept hypothesis H2.

4.4. Effect of Irony

Figure 4c shows the means for matching (black), mismatching (dark grey) and neutral utterances (light grey). There is a clear shift of the mismatching utterances towards the neutral area in the evaluation space. This shift was significant (Tukey post-hoc test $p<0.05$) for Lax voice quality.

Looking more closely at the distributions of these five categories (See Figure 5), we can see a marked difference between matching and mismatching/neutral utterances.

Subjects responses are constrained to be within (or almost within) the activation/evaluation circle. This puts a limits on possible differences in variance by constraining the tails of the distributions. This results in skew for distributions with off centre means, hence the different distribution shapes of non-ambiguous (matching) utterances, from ambiguous (mismatching) utterances and neutral utterances.

Overall we see a significant shift towards neutral valence for the Lax/-ve utterances and a non-significant tendency for Str/+ve utterances to also move towards neutral valence. We have already shown that both voice quality and sentence content affect valence, therefore we can conclude that we have succeeded in creating ambiguous utterances where contrasting features are creating an element of tension allowing us to accept hypothesis H3.

5. CONCLUSION

We have shown that the use of the activation/evaluation space is a useful and effective means of evaluating valence shifts caused by competing features in spoken utterances. We have also shown the impact of voice quality on perceived emotion in terms of valence and activation and how the text content of the utterance also modifies the perception of valence.

Furthermore, a combination of voice quality associated with positive valence and text content associated with negative valence creates a mismatch which produces a perceived valence closer to the neutral part of the scale. As we have significant evidence that these

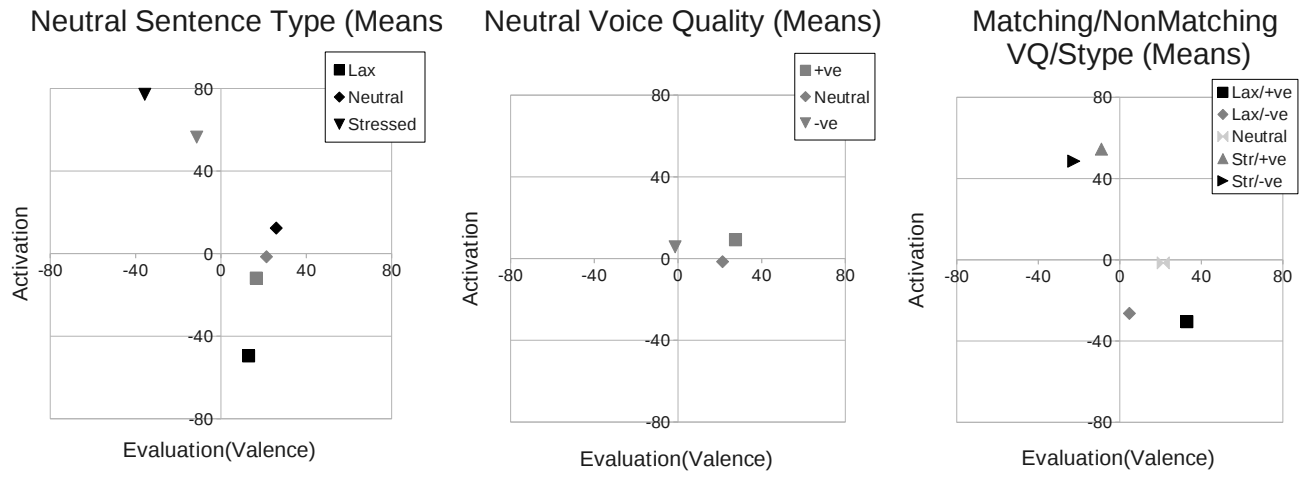


Fig. 4. a) Mean values for neutral sentences by different voice qualities. Black - Natural Speech, Grey - Synthetic Speech. b) Mean values by text content for neutral synthesised utterances. Positive sentences - '+ve', negative sentences - '-ve'. c) Mean values by matching/nonambiguous (black), mismatching/ambiguous (dark grey) and neutral (light grey) synthesised utterances.

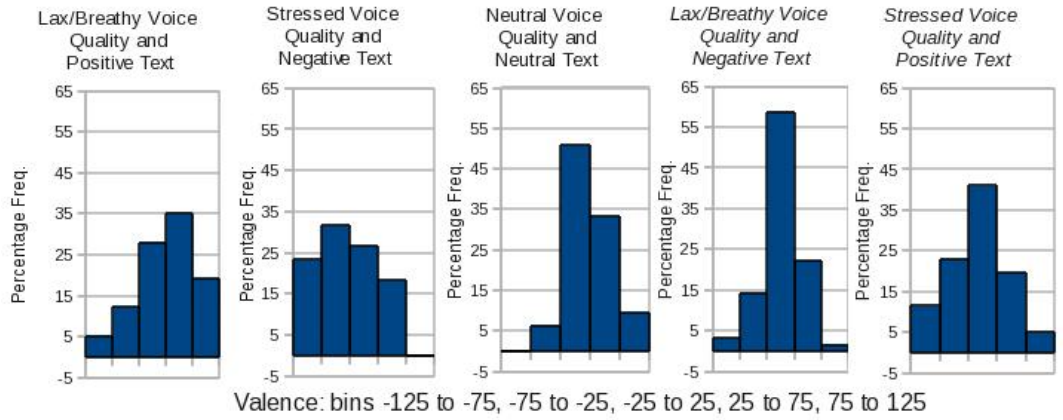


Fig. 5. Comparison of the distribution of valence results. **TCONT/VQ** mismatching/ambiguous utterances in italics, (left). matching/non-ambiguous conditions (right) and responses to neutral condition (centre).

features produce an effect on valence in isolation, together they are creating a tension in the utterance and producing an emotionally ambiguous stimuli.

However, a limitation of our activation/evaluation space approach is that it can't distinguish between a neutral utterance and an ambiguous one. Subjects were asked to give a single response and Figure 5 shows that conflicting features do not create a bi-modal response but are instead merged.

However, qualitatively the mismatched utterances do not sound like the neutral utterances. We have made an example of all nine conditions available on the internet [1] and encourage the reader to listen to the differences.

Although this is strong indirect evidence of creating ambiguous utterances, we would like to have a more explicit way of testing the difference between the neutral and the ambiguous. This requires more advanced evaluation methodologies which can deal with issues such as motivation, intention and conversational function. We believe the results presented here offer a good starting point for investigating these higher level responses to synthetic speech stimuli. Future work will investigate the direct affect of ambiguity on the perception of character, and, through the assessment of more complete systems, the utility of this approach in increasing engagement.

[1] http://homepages.inf.ed.ac.uk/matthewa/ssw2013VQ/

6. ACKNOWLEDGEMENTS

This research was funded by the Royal Society through a Royal Society Industrial Fellowship.

7. REFERENCES

[1] A. J. Morrison, P. Mitchell, and M. Brereton, "The lens of ludic engagement: evaluating participation in interactive art installations," in *Proceedings of the 15th international conference on Multimedia*, ser. MULTIMEDIA '07, 2007, pp. 509–512.

[2] J. McCarthy and P. Wright, *Technology as Experience*. MIT Press, 2004.

[3] W. W. Gaver, J. Beaver, and S. Benford, "Ambiguity as a resource for design," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ser. CHI '03, 2003, pp. 233–240.

[4] American Heritage, *The American Heritage Dictionary of the English Language.*, 4th ed. Houghton Mifflin Company, 2009.

[5] M. Scrhöder, "Emotional speech synthesis: A review." in *Proceedings Eurospeech 01*, 2001, pp. 561–4.

[6] C. Gobl and A. N. Chasaide, "The role of voice quality in communicating emotion, mood and attitude," *Speech Communication*, vol. 40, no. 1-2, pp. 189–212, Apr. 2003.

[7] M. Scrhöder and M. Grice, "Expressing vocal effort in concatenative synthesis." in *ICPhS*, 2003, pp. 2589–92.

[8] M. Aylett and C. Pidcock, "Adding and controlling emotion in synthesised speech," UK Patent GB2 447 263A, September 10, 2008.

[9] G. Hofer, K. Richmond, and R. Clark, "Informed blending of databases for emotional speech synthesis," in *Proc. Interspeech*, 2005.

[10] M. P. Aylett and C. J. Pidcock, "The cerevoice characterful speech synthesiser sdk," in *AISB*, 2007, pp. 174–8.

[11] R. Cowie, E. Douglas-Cowie, S. Savvidou, E. McMahon, M. M. Sawey, and M. Scrhöder, "Feeltrace': An instrument for recording perceived emotion in real time." in *ISCA Workshop on Speech and Emotion*, 2000, pp. 19–24.

[12] H. Schlosberg, "A scale for judgement of facial expressions." *Journal of Experimental Psychology*, vol. 29, pp. 497–510, 1954.

[13] R. Plutchik, *The Psychology and Biology of Emotion.* New York: Harper Collinns, 1994.

[14] C. M. Whissell, "The dictionary of affect and language." in *Emotion: Theory, Research, and Experience.*, R. Plutchik and H. Kellerman, Eds. Academic Press, 1989, vol. 4, pp. 113–131.

[15] S. King and V. Karaiskos, "The blizzard challenge 2010," in *Blizzard Challenge Workshop*, 2010.

Interactional Adequacy as a Factor in the Perception of Synthesized Speech

Timo Baumann

Department of Informatics
Universität Hamburg, Germany
baumann@informatik.uni-hamburg.de

David Schlangen

Faculty of Linguistics and Literature
Bielefeld University, Germany
david.schlangen@uni-bielefeld.de

Abstract

Speaking as part of a conversation is different from reading out aloud. Speech synthesis systems, however, are typically developed using assumptions (at least implicitly) that are more true of the latter than the former situation. We address one particular aspect, which is the assumption that a fully formulated sentence is available for synthesis. We have built a system that does not make this assumption but rather can synthesize speech given incrementally extended input. In an evaluation experiment, we found that in a dynamic domain where what is talked about changes quickly, subjects rated the output of this system as more 'naturally pronounced' than that of a baseline system that employed standard synthesis, despite the synthesis quality objectively being degraded. Our results highlight the importance of considering a synthesizer's ability to support interactive use-cases when determining the adequacy of synthesized speech.
Index Terms: speech synthesis, incremental processing, interactive behaviour, evaluation, adequacy

1. Introduction

Most speech synthesis software is not tailored towards interactive use, but instead operates in a way that is best described as reading out aloud. As a consequence, full sentences (or utterances in dialogue) are used as input units, and typically, input cannot be changed or extended after the synthesizer's processing has started.

This coarse input granularity and monolithical processing reduce the ability to adapt to unforeseen changes in the environment, which may be necessary (or at least advantageous) in interactive systems, such as commentary generation, or conversational dialogue systems. Thus, interactive systems may profit from speech synthesis that uses smaller, partial input units that are extended *incrementally* and just-in-time, while speech output is already ongoing, to produce an utterance in a piece-meal fashion.

Dutoit et al. [1] have previously shown that incremental, HMM-based speech synthesis is possible and only moderately degrades synthesis quality; however, their speech synthesizer is not integrated into a full text-to-speech system. We have built an interactive text-to-speech synthesizer, INPRO_iSS [2],[1] based on MaryTTS [3] and the incremental processing toolkit INPROTK [4], which is able to produce output based on incrementally expanded utterance descriptions, and which also allows to change delivery parameters of ongoing speech, such as tempo, pitch, and – added in this work – force.

We have previously shown that incremental speech synthesis, in combination with incremental natural language generation, is profitable in order to remain flexible with regards to external events from the environment [5]. In a user study, participants rated the naturalness of the formulation and the pronunciation of our system in a highly dynamic environment. Analysis of participant ratings showed that the formulations enabled by incrementally synthesizing speech were preferred (by a large margin) over baseline formulations, even if incremental formulation sometimes has to resort to using a hesitation when events unfold more slowly than anticipated [6]. In this work, we present the result that users in addition rated the incremental system's pronunciation as significantly more natural, despite that fact that objectively pronunciation quality was lower. In our opinion, this result highlights the importance of considering a synthesizer's abilities to support interactive use-cases when determining the 'quality' of the synthesized speech.

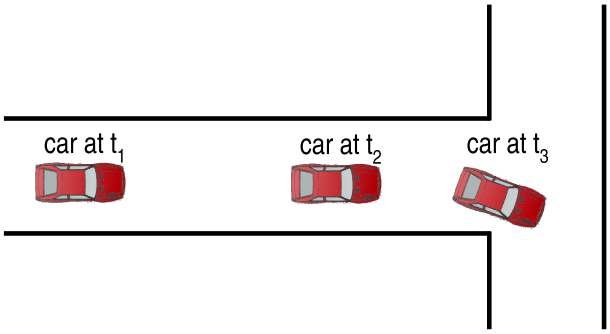
In Section 2, we detail our system's implementation for incrementally provided input as well as timely adaptation of delivery parameters. We describe the domain of our system in Section 3, the evaluation experiment in Section 4, and present the results in Section 5. We draw conclusions from the experiment in Section 6 and outline ideas for future work in Section 7.

2. Incrementality and timely adaptation

In our system, textual material to be synthesized is added in 'chunks', which ideally correspond to phrases, but which may also be shorter, down to individual words. Chunks are added to the system incrementally, and prosody is re-computed to reflect changes in the textual and prosodic analysis given the added material as soon as the material becomes available. This means that prosodic quality is highest when material is added early on, but our previous work has shown that having one chunk/phrase of lookahead at all times is sufficient for prosody (pitch and duration) to be almost indistinguishable to non-incrementally produced pitch and duration assignments [7]. It is also possible to revoke parts of the input (that have not been produced yet), and to construct *utterance plans* [8], which may contain multiple alternative paths for possible realization that can be selected until immediately before speech realization reaches the branching point in the plan.

Incremental extension of ongoing utterances allows the system to generate behaviour such as the one shown in Figure 1: in the figure, a car is shown driving along a street, and eventually turning. An incremental system that is to comment on these

[1] INPRO_iSS is available as part of the INPROTK distribution at http://inprotk.sf.net.



time	event description	ongoing utterance (already realized part in **bold**, newly appended continuation in *italic*)
t_1	car on Main Street	**The** car drives along Main Street.
t_2	car will likely turn	. . . **drives along Main Stre**et *and then turns ‹hes›*
t_3	car turns right	. . . **drives along Main Street and the**n turns *right.*

Figure 1: Example of incremental utterance production as a car drives along a street and turns. The ongoing utterance is extended as information becomes available.

events is able to generate one complex, successively extended utterance, as in the figure, by adapting ongoing synthesis. As in the figure, the system may hypothesize the upcoming turn at time t_2 and start to output the part of the utterance that is independent of the direction of the turn. It may then speak about the direction of the car's turn immediately when it happens at t_3. In contrast, a non-incremental system has to wait until t_3 before it may start its commentary about the car turning because it requires to know the direction of the turn, despite of the fact that the car *will* likely turn was known at time t_2 and the beginning of the description ("and then turns") being identical for either direction. Of course, an incremental system may mis-judge the time at which the direction of the car turning (or any other anticipated event) happens. As a countermeasure, our system may be ordered to output a hesitation when it runs out of speech material, in order to gain time (as shown in the second line of the example in Figure 1). Hesitations are skipped (or immediately aborted) as soon as more speech material becomes available (as shown in the third line of the example).

The architectural overview of our system, as given in Figure 2, shows the *just-in-time* approach that is used. The overall goal is to perform processing steps as late as possible, which keeps overheads that are due to later changes of the input to a minimum. In addition, most of the processing time is moved into the *delivery time* of the speech, resulting in improved system response compared to standard processing (see also [5]). The time at which processing is required depends on the level of abstraction: vocoding need only be performed immediately before the corresponding audio is requested, and HMM optimization is performed step-wise using local phoneme contexts (as proposed by [1], but also using global variance optimization [9] within the local context) for each phoneme; higher-level processing must be performed somewhat in advance, and needs to be able to accomodate changes that may result from later addition/revocation of input.

Our processing architecture INPROTK [4] is based on *incremental units* (IUs) [10]. IUs are shown as boxes in Figure 2 and related units are connected via *same level links* for data of

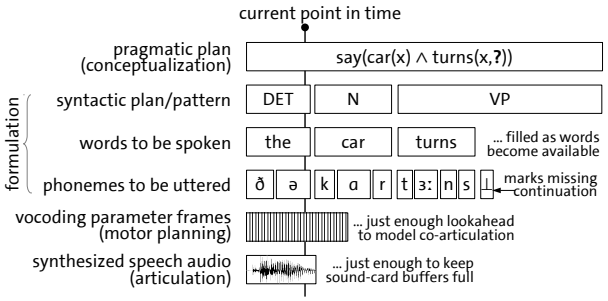


Figure 2: Hierarchical structure of incremental units describing an example utterance as it is being produced during utterance delivery.

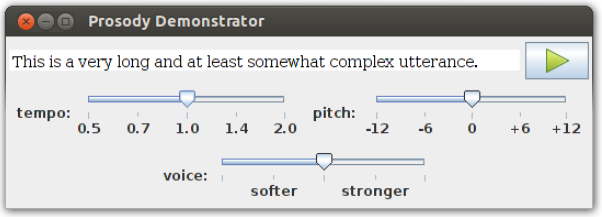


Figure 3: Example graphical interface to incrementally manipulate speech delivery parameters.

the same type (shown in the figure by horizontal alignment) and *grounding links* for hierarchical dependence over different levels (shown in the figure by placing units above/below other units. The links are used to track dependencies in the system and both links and units are revised whenever material is added, removed, or changed incrementally. Furthermore, IUs are active objects, which are set up to automatically request relevant processing steps via an update mechanism. Linguistic pre-processing and prosody assignment relies on MaryTTS [3], which is called repeatedly whenever new material is added to the ongoing utterance.

Linguistic pre-processing and (to a lesser degree) HMM optimization are computationally expensive. For this reason, we added ways to adapt speech delivery parameters outside of the HMM framework that work with almost zero delay. The system uses STRAIGHT vocoding [11], and is able to alter the different vocoding parameters (pitch, cepstrum, energy, and voicing strengths) until immediately before a frame is vocoded. Furthermore, to allow for a simple, yet effective method to change speech tempo without requiring to reperform the HMM optimization, we allow the system to skip or to repeat generated parameter frames, which leads to faster (or slower, respectively) speech – however, ignoring the HMM optimality criteria. (It should be noted that this method works well for moderate tempo changes ($\pm 30\%$) only and leads to acoustic artifacts for extreme changes.)

The capabilities of our adaptation method are exploited in a demonstrator, depicted in Figure 3: it allows to alter pitch, tempo, and voice force (a linear combination of changes in total energy, spectral tilt, as proposed in [12], and additionally voicing strength) in real time (less than 5 ms delay). However, this capability is used only to a limited degree in the experiment

reported below (pitch and duration are adapted just-in-time in the vicinity of hesitations).

3. System domain

To test the merit of incremental speech synthesis, we built a system for an interactive commentary domain. The domain combines aspects of sports commentary [13], which often profits from open-ended utterances, with interactive map exploration descriptions for the visually impaired [14].

In our *CarChase* domain, shown in Figure 4, a car drives around the streets on the map and a commentator (supposed to be observing the scene from above) comments on where it is driving and what turns it is taking.

The car's itinerary in our domain simulator is scripted from a configuration file which assigns target positions for the car at different points in time and from which the motion and rotation of the car is animated. The speed of the car is set so that the event density is high enough that the setting cannot be described by simply producing one utterance per event; instead, utterances need to be aborted to make room for new material (baseline behaviour), or utterances need to integrate later events while they are already ongoing (incremental behaviour).

Our system distinguishes three different types of events: street *identification*, the car taking a *turn*, and *turn preparatory* events that become active when it is obvious that the car will turn but the direction of the turn cannot yet be determined. The three event types are shown in Figure 1 at times t_1 (*ID*), t_2 (*turn-prep*), and t_3 (*turn*). While it is an advantage of the incremental system that it may combine multiple events into one longer, connected utterance, the main advantage for temporal adequacy of the commentary comes from *turn-prep* events, which allow to start producing some material about the event (the fact that a turn will occur) even before the direction of the turn can be specified.

The focus of our work is only on incremental speech synthesis, and hence we did not implement an automatic scene analysis/event detection nor an NLG component for the task (however, see [15, 16] for such components in a highly related domain). Instead, commentary text is scripted from the same configuration file that controls the car's motion on the board. Events that control speech synthesis lag behind motion events slightly, ensuring that visual analysis would be possible, and event/text correspondence – although hand-written – is close, matching NLG capabilities.

4. Experiment

We evaluated the incremental system by comparing its output to a non-incremental baseline system which is unable to extend ongoing partial utterances and hence cannot incrementally combine multiple events into one utterance. Instead, the baseline system produces one full utterance per event. To ensure timeliness of commentary even in the baseline system, some commenting events were marked as optional (in which case the corresponding utterances are skipped if the system is still outputting a previous utterance), whereas non-optional utterances abort any ongoing commentary in favour of the next utterance. All *turn* events in the domain were marked as optional, all street *ID* events as non-optional. Of course, the baseline system cannot make use of *turn-prep* events.

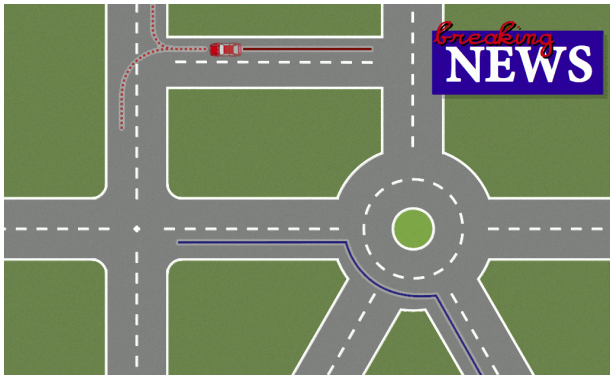


Figure 4: The map shown in the *CarChase* domain, including the car on one of its itineraries (red). At the depicted moment we can assume that the car will take a turn, but do not know whether left or right. A second itinerary is shown in blue.

We devised 4 different configurations (including the itineraries shown in Figure 4), and the timing of events was varied (by having the car go at different speeds, or by delaying some events), resulting in 9 scenarios; in 3 of these, the incremental system *over-commits* to the appearance of a *turn* event and needs to play a short hesitation ('ehm') before the direction of the turn event becomes known. These cases were meant to include errors that are specific to the incremental system's behaviour into the evaluation and thus lead to a more balanced comparison to the baseline system.

Both systems' output for the 9 scenarios was recorded with a screen recorder, resulting in 18 videos that were played in random order to 9 participants (university students not involved in the research) who were told that various versions of commentary-generating systems generated the commentary based on the running picture in the videos and were then asked to rate each video on a five-point Likert scale with regards to how natural (similar to a human) the spoken commentary was (a) formulated, and (b) pronounced. We did not further specify what exactly was meant by 'formulation' or 'pronunciation', instead relying on the participants' intuitive understanding of these terms. In total, the questionnaires resulted in 81 paired samples for each question.

The experiment was performed with an early version of the system, which still performed some prosodic mis-alignments at utterance extensions, due to various shortcomings. Furthermore, the coarsely implemented hesitations result in audible acoustic and prosodic artifacts. Overall, we hoped that the incremental system's formulation would be preferred by participants, without a significant decrease in pronunciation ratings.

5. Results

As expected and shown in Figure 5, participants highly preferred the incremental system's formulations over the non-incremental baseline system, with a median difference in ratings of the two conditions of 2 points (mean 1.66), which is highly significant (sign test, 68+/9=/4-; $p < .0001$). For the incremental system, we distinguished between settings where the system generated a hesitation (*hes*) and those where it did not (*no hes*). As can be seen in the figure, even utterances in which the incremental sys-

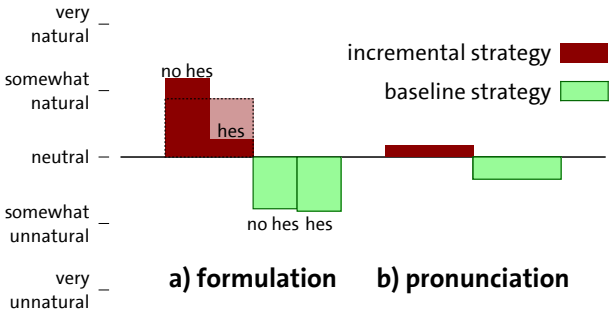


Figure 5: Mean ratings of formulation and pronunciation quality for the incremental and the baseline system. The formulation rating is shown subdivided for utterances with and without hesitations.

tem had to resort to a hesitation were rated as significantly better formulated than the baseline behaviour (see also [6]). There was no significant difference between pronunciation ratings for the *hes*/*no hes* conditions.

More relevant for the present discussion, however, are the *pronunciation* rating differences between the incremental and baseline systems, which also show a clear preference for the incremental system, with a mean difference in ratings of the two conditions of 0.51 points, which was also highly significant (sign test, 38+/30=/13-; $p < .0007$).[2]

The better pronunciation ratings are especially surprising, as objectively, the synthesis quality of the incremental system can only have been systematically lower than that of the non-incremental system, as the manipulations to synthesis required for incremental processing (and the flaws that existed in the early prototype that was used in the experiment) can only systematically result in a deterioration of the synthesis quality, but not in a systematic improvement. Thus, it appears that participants pardoned bad *synthesis quality* (which occurs in both system versions for certain words) more easily, when overall *formulation quality* is better and even compensate for hesitations that may have been realized rather unnaturally in the incremental system. More to the point: naïve participants do not clearly distinguish between pronunciation and formulation ratings (this is also evidenced by the fact that ratings for the two questions are moderately correlated; Pearson's $r = .537$), and formulation seems to outweigh pronunciation.

Of course, applied systems are most often used by naïve users. Thus, their ratings should matter much more than objective metrics or ratings given by professionals.

6. Conclusion

We have built an incremental speech synthesis system that accepts incrementally provided input and we tested it in a domain where this capability allows to integrate multiple, successive events into one complex utterance, and – using preparatory events – allows very timely behaviour. Our experiment shows that the incremental system's formulations are highly preferred over conventional baseline behaviour, even when they involve the introduction of (poorly synthesized) hesitations.

Furthermore, the incremental system's synthesis quality (as captured by the pronunciation rating) was rated as significantly better, despite of modifications that can only have lead to objectively lower quality. However, the speech that was synthesized incrementally was interactionally more *adequate* to the situation of continuous commentary, that is, there were other aspects than voice quality that mattered to the perception of the synthesized speech.

We conclude that synthesis quality may actually matter very little in comparison to *interaction quality*, and that speech synthesis systems should be evaluated in context, or at least taking into account the sorts of interaction behaviour that they support (such as incremental behaviour in our case). In the end, interactive adequacy as a target of speech synthesis optimization may lead to better results more easily than (isolated) perception ratings of synthesized speech samples, without their integration into the relevant context.

Similarly to spoken commentary in a dynamic domain as presented above, conversational speech requires revisions and reactions to external events, such as listener feedback (or the absence thereof) [17, 18]. Thus, we believe that our results, as well as incremental processing in general, also apply to a broad range of conversational synthesis tasks. Finally, the ability to adapt distinguishes incremental speech synthesis from canned speech, which may sound better (seen in isolation), but is completely static and unresponsive to situational demands. Thus, the current success of canned speech in dialogue systems cannot be expected to scale to more interactively advanced dialogue in conversational settings.

7. Future work

Our current system is a combination of incremental (vocoding, HMM optimization, top-level integration) and non-incremental strategies (linguistic pre-processing, HMM state selection), which is a compromise owing to the complexity of the full text-to-speech task. However, we plan to extend our system, which is already available as open-source software, to model more of the (phrasal) structure that is generated by linguistic preprocessing in the incremental data structure (cmp. Figure 2). This will allow to e. g. support SSML as incremental input (which is currently unsupported), to support the structured, high-level manipulation of the prosodic realization in real time (i. e. without further re-processing), and allow for a flexible blend of text-to-speech and concept-to-speech techniques in the incremental system.

Modelling higher-level structure will also include modelling *underspecified* higher-level structure, for example the fact that a question is to be synthesized (triggering the appropriate sentence intonation) despite the fact that some specific content is still unknown. In general, there is a trade-off between early specification, and the likelihood of later revision; quality of the system output might improve with explicit models of such likelihoods and corresponding processing adaptations.

Acknowledgements The first author would like to thank Petra Wagner and Wolfgang Menzel for fruitful discussions on the topic, and permanent encouragement.

[2] We also conducted a non-paired, two-tailed t-test for pronunciation ratings, as the different formulations of the systems might have effects on pronunciation quality; this test was also significant ($p < .0012$).

8. References

[1] T. Dutoit, M. Astrinaki, O. Babacan, N. d'Alessandro, and B. Picart, "pHTS for Max/MSP: A streaming architecture for statistical parametric speech synthesis," Université de Mons, Tech. Rep. 1, 3 2011. [Online]. Available: http://www.numediart.org/docs/numediart_2011_s13_p2_report.pdf

[2] T. Baumann and D. Schlangen, "INPRO_iSS: A component for just-in-time incremental speech synthesis," in *Procs. of ACL System Demonstrations*, Jeju, Korea, 2012.

[3] M. Schröder and J. Trouvain, "The German text-to-speech synthesis system MARY: A tool for research, development and teaching," *International Journal of Speech Technology*, vol. 6, no. 3, pp. 365–377, Oct. 2003.

[4] T. Baumann and D. Schlangen, "The INPROTK 2012 release," in *Proceedings of SDCTD*, Montréal, Canada, 2012.

[5] H. Buschmeier, T. Baumann, B. Dorsch, S. Kopp, and D. Schlangen, "Combining incremental language generation and incremental speech synthesis for adaptive information presentation," in *Proceedings of SigDial*, Seoul, Korea, 2012, pp. 295–303.

[6] T. Baumann and D. Schlangen, "Open-ended, extensible system utterances are preferred, even if they require filled pauses," in *Proceedings of SigDIAL*, Metz, France, Sep. 2013.

[7] T. Baumann and D. Schlangen, "Evaluating prosodic processing for incremental speech synthesis," in *Proceedings of Interspeech*. Portland, USA: ISCA, Sep. 2012.

[8] G. Skantze and A. Hjalmarsson, "Towards incremental speech generation in dialogue systems," in *Proceedings of SIGdial*, Tokyo, Japan, Sep. 2010.

[9] T. Toda and K. Tokuda, "A speech parameter generation algorithm considering global variance for hmm-based speech synthesis," *IEICE transactions on information and systems*, vol. 90, no. 5, pp. 816–824, 2007.

[10] D. Schlangen and G. Skantze, "A General, Abstract Model of Incremental Dialogue Processing," in *Proceedings of the EACL*, Athens, Greece, 2009, pp. 710–718.

[11] H. Kawahara, "Speech representation and transformation using adaptive interpolation of weighted spectrum: vocoder revisited," in *Acoustics, Speech, and Signal Processing, 1997. ICASSP-97., IEEE International Conference on*, vol. 2. IEEE, 1997, pp. 1303–1306.

[12] N. Ström and S. Seneff, "Intelligent barge-in in conversational systems," in *International Conference on Spoken Language Processing (ICSLP)*, Beijing, China, Oct. 2000.

[13] D. L. Chen and R. J. Mooney, "Learning to sportscast: A test of grounded language acquisition," in *Proceedings of 25th International Conference on Machine Learning (ICML-2008)*, Helsinki, Finland, Jul. 2008.

[14] K. Lohmann, M. Kerzel, and C. Habel, "Verbally assisted virtual-environment tactile maps: A prototype system," in *Proceedings of the Workshop on Spatial Knowledge Acquisition with Limited Information Displays 2012*, C. Graf, N. A. Giudice, and F. Schmid, Eds., 2012, pp. 25–30.

[15] M. Kerzel and C. Habel, "Monitoring and describing events for virtual-environment tactile-map exploration," in *Proceedings of Workshop on 'Identifying Objects, Processes and Events', 10th International Conference on Spatial Information Theory*, A. Galton, M. Worboys, and M. Duckham, Eds., 2011, pp. 13–18.

[16] K. Lohmann, O. Eichhorn, and T. Baumann, "Generating situated assisting utterances to facilitate tactile-map understanding: A prototype system," in *Proceedings of SLPAT 2012*, Montreal, Canada, 2012.

[17] H. H. Clark, *Using Language*. Cambridge University Press, 1996.

[18] H. H. Clark, "Speaking in time," *Speech Communication*, vol. 36, no. 1, pp. 5–13, 2002.

A novel irregular voice model for HMM-based speech synthesis

Tamás Gábor Csapó, Géza Németh

Department of Telecommunications and Media Informatics,
Budapest University of Technology and Economics, Budapest, Hungary
{csapot,nemeth}@tmit.bme.hu

Abstract

State-of-the-art text-to-speech (TTS) synthesis is often based on statistical parametric methods. Particular attention is paid to hidden Markov model (HMM) based text-to-speech synthesis. HMM-TTS is optimized for ideal voices and may not produce high quality synthesized speech with voices having frequent non-ideal phonation. Such a voice quality is irregular phonation (also called as glottalization), which occurs frequently among healthy speakers. There are existing methods for transforming regular (also called as modal) to irregular voice, but only initial experiments have been conducted for statistical parametric speech synthesis with a glottalization model. In this paper we extend our previous residual codebook based excitation model with irregular voice modeling. The proposed model applies three heuristics, which were proven to be useful: 1) pitch halving, 2) pitch-synchronous residual modulation with periods multiplied by random scaling factors and 3) spectral distortion. In a perception test the extended HMM-TTS produced speech that is more similar to the original speaker than the baseline system. An acoustic experiment found the output of the model to be similar to original irregular speech in terms of several parameters. Applications of the model may include expressive statistical parametric speech synthesis and the creation of personalized voices.

Index Terms: irregular phonation, glottalization, voice quality, parametric, speech synthesis

1. Introduction

State-of-the art text-to-speech (TTS) synthesis is often based on statistical parametric methods. Particular attention is paid to hidden Markov model (HMM) based text-to-speech synthesis [1] (HTS). In this type of speech synthesis, the speech signal is decomposed to physical parameters which are fed to a machine learning system. After the training data is learned, during synthesis, the parameter sequences are converted back to speech signal with speech coding methods. For this task, often simple vocoders (e.g. pulse-noise excitation) are used which make use of the source-filter model of speech. The advantages of HMM-TTS compared to other synthesis techniques include its flexibility and small footprint.

However, the over-simplified vocoder techniques make the quality of synthesized speech of HMM-TTS poor compared to high-quality unit selection based speech synthesis systems. To overcome this drawback, several improved excitation models have been proposed. STRAIGHT-based vocoding produces very good quality HMM-based synthesized speech [2]. Cabral uses the Liljencrants-Fant (LF) [3] acoustic model of the glottal source derivative to construct the excitation signal [4]. Drugman proposed the Deterministic Plus Stochastic Model (DSM) of the residual signal [5]. Raitio and his colleagues use glottal inverse filtering within HMM-based speech synthesis and unit selection of pulses for generating natural sounding synthetic speech [6], [7]. The latest excitation models introduce the voicing cut-off frequency [8] and waveform interpolation [9] to enhance the performance of HMM-TTS. We proposed a residual codebook based excitation model which also exceeds the quality of simple pulse-noise excitation [10], [11].

1.1. Irregular phonation

Statistical parametric speech synthesis and most of the above excitation models are optimized for ideal voices and may not produce high quality synthesized speech with voices having frequent non-ideal phonation. Such a non-ideal voice quality is irregular phonation.

During regular voiced phonation (ideal, modal voice) in human speech, the vocal cords are vibrating quasi-periodically. For shorter or longer periods of time this vibration may become irregular. Abrupt changes occur in the fundamental frequency (F0), amplitude of the pitch periods or both. This is called irregular phonation (or glottalization, vocal fry, creaky voice), which is a frequent phenomenon for both healthy speakers and people having voice disorders. It is often accompanied by extremely low pitch and the quick attenuation of glottal pulses. Glottalization is perceived as a creaky, rough voice [12], [13]. Fig. 1 shows an example for glottalization (LP residual on the top and speech signal on the bottom). The horizontal arrow denotes the section where the phonation is irregular. Amplitude attenuations in the waveform and missing impulses in the residual are clearly visible.

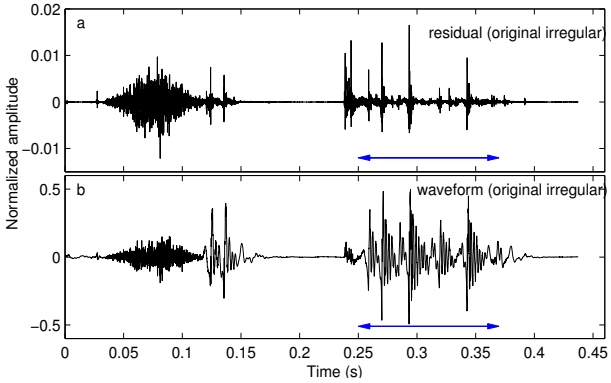


Figure 1: *A speech recording of the word 'cipő' having irregular phonation at the section denoted by an arrow. a) residual signal and b) speech signal.*

It was found that up to 15% of the vowels of healthy American English speakers may be produced with irregular phonation [14]; therefore it is not negligible in normal speech. The occurrence of glottalization depends on the prosodic structure (it often coincides with prosodic boundaries and stressed syllables [15]) and carries information from the speaker, his/her dialect, mood and emotional state [16]. Irregular phonation can cause problems for standard speech analysis methods (e.g. F0 tracking and spectral analysis). Proper modeling of irregularly phonated speech may

contribute to building natural, emotional and personalized speech synthesis systems. Irregular phonation is frequently adopted in lively story-telling, natural interactive conversation [17] and can signal sadness [18] or boredom [19]. Therefore an irregular phonation model improves expressive speech synthesis systems. Such a model allows speaker adaptation for deep elderly voices (e.g. radio announcers) having frequent glottalization.

First attempts to model irregular phonation were either in the formant synthesis domain [20] or relied on increasing jitter and shimmer of the speech signal [21]. In [13], a simple semi-automatic transformation method is developed which introduces irregular pitch periods into a modal speech signal, based on amplitude scaling of the individual periods. In perception and acoustic experiments, this method was shown to yield irregular speech that is as rough and as natural as original glottalized speech. To model vocal fry in statistical parametric speech synthesis, [22] introduces a robust F0 measure and two-band voicing, which improves significantly the quality of HMM-based speech synthesis. However, they do not focus on the characteristics of creaky excitation. Drugman and his colleagues derive an extension of the DSM model [5] which can handle creaky excitation by integrating secondary pulses in the residual, and investigate this in copy-synthesis experiments [23]. After that they investigate the usefulness of contextual factors for creaky voice prediction and experiment with adding parameter streams describing irregular phonation into the HMM-TTS framework [17]. To the best of our knowledge this extended analysis-synthesis method with the creaky voice model has not been integrated into HTS yet.

In this paper we extend our previous residual codebook based excitation model (HTS-CDBK) with irregular voice modeling. The baseline residual analysis-synthesis framework and the model of irregular voice are introduced in Sections 2 and 3, respectively. In Section 4 a perceptual test, while in Section 5 an acoustic experiment and their results are shown. In Section 6, we present the advantages and drawbacks of our method and conclude the paper.

2. HMM-TTS with a residual codebook based excitation model

We have proposed a residual codebook based excitation model [10] and integrated it into HMM-TTS ([11], HTS-CDBK), that will be used here as the baseline system.

2.1. Analysis

The input is a speech waveform with 16 kHz sampling rate and 16 bit linear PCM quantization. First, the F0 parameters are calculated by the publicly available Snack pitch tracker with 25 ms frame size and 5 ms frame shift. In the next step 34-dimensional MGC analysis is performed on the speech signal with the SPTK tool. The residual signal (excitation) is obtained by MGLSA inverse filtering. Next, a Glottal Closure Instant (GCI) detection algorithm is used to find the pitch boundaries in the voiced parts of the modal speech signal [24]. Finally, a codebook of pitch-synchronous residuals is built, obtained from a small speech database (see Section 2.4) and residual analysis is performed.

The further analysis steps are completed on the residual signal with the same frame shift values. For measuring the parameters in the voiced parts, pitch synchronous, two period long frames are used according to the GCI locations and they are Hanning-windowed (see Fig. 2). A codebook is built from pitch-synchronous residual frames. Several parameters of these frames are used to fully describe the speech residuals:

- F0: fundamental frequency of the frame
- gain: RMS energy of the windowed frame
- rt0 peak indices: the locations of prominent values (peaks or valleys) in the windowed frame (see Fig. 2)
- HNR: Harmonic-To-Noise ratio of the frame [25]

For each voiced frame, one codebook element is saved with the above parameters and the windowed signal is also stored. The rt0 parameter is a 4-dimensional vector, which is a new idea for describing the residual frames. We found that the consecutive rt0 parameters are slowly evolving enough and are suitable for machine learning in HTS. In the used parameters our model is different from similar excitation models, like DSM [5]. These parameters will be used for target cost calculations during synthesis. In order to collect similar codebook elements, the RMSE distance is calculated between the pitch normalized versions of the codebook elements which will be used for concatenation cost. The normalization is done by resampling every frame to 40 samples.

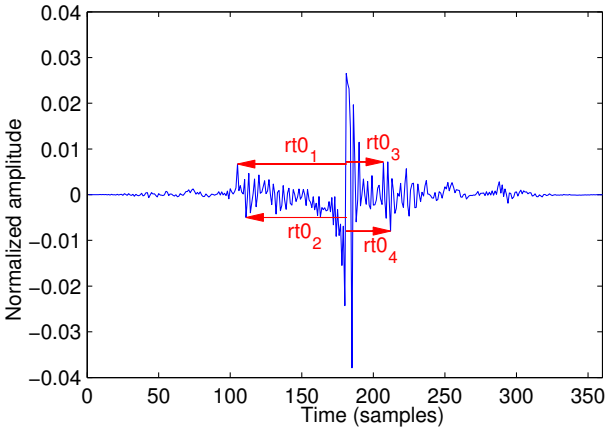


Figure 2: *Calculation of the rt0 parameter for a windowed residual segment. $rt0_i$ is the distance of prominent peaks from the main impulse, in samples.*

2.2. Training

For training, the parameters of MGC, log(F0), log(gain), log(rt0) and log(HNR) of each frame are extracted. F0 and rt0 are modeled with MSD-HMMs because these do not have values in unvoiced regions. MGC, HNR and gain are modeled as simple HMMs. The first and second derivatives of all of the parameters are also stored in the parameter files and used in the training phase. Altogether five streams of data are considered.

2.3. Synthesis

In the synthesis phase of HTS-CDBK the inputs are the parameters obtained during training (F0, gain, rt0 indices and HNR) and the codebook of pitch-synchronous residuals. If the frame is voiced, a suitable codebook element with the target F0, rt0 and HNR is searched from the codebook. We apply target cost and concatenation cost with hand-crafted weights, similarly to unit selection speech synthesis [26]. The target cost is the squared difference among the parameters (F0, rt0 and HNR) of the current frame and the parameters of those elements in the codebook. The concatenation cost is calculated as the RMSE difference of the pitch normalized frames. When a suitable codebook element is found, its fundamental period

is set to the target F0 by either zero padding or deletion. If the frame is unvoiced, white noise is used as excitation. Next, the residual is created by pitch synchronously overlap-adding the Hanning-windowed residual periods. After that, the synthesized residual is lowpass filtered to 6 kHz and white noise is used in the frequency band above 6 kHz. Finally, the energy of the frames is set using the gain parameter and synthesized speech is reconstructed by MGLSA filtering.

Note that the computational cost of the residual unit selection during synthesis depends on the size of the codebook and the applied costs. In our experiments we found that using a small codebook the synthesis time might be suitable for real-time synthesis, therefore the method does not decrease the flexibility of the original HTS system.

2.4. Speech data

The speech data that was used for our experiments is a part of the PPBA database [27]. Two Hungarian males were chosen for speaker dependent training (denoted FF3 and FF4). Both speakers produced irregular phonation frequently, mostly at the end of sentences. 1940-1940 phonetically balanced sentences (2-2 hours of speech) from them were used as training corpora. The sentences in the corpus are stored as waveform files (44.1 kHz sampling rate, 16 bit linear PCM quantization), which were resampled to 16 kHz. We created a residual codebook with 3394 elements for speaker FF3 and another one with 2218 elements for speaker FF4 extracted from about 10 minutes of speech from the first 150 sentences. Other excitation models use codebooks of similar scale [7].

2.5. Irregular voice handling in the baseline system

We have analyzed the training speech databases of the two speakers and conducted speaker dependent training. During the analysis, it was found that when glottalization occurs (typically in the vowels of the last syllables of the sentences), the Snack pitch tracker cannot measure F0 and sets the frame as being unvoiced. Therefore, this pattern is learned by the system and glottalization is modeled in HTS-CDBK similarly to unvoiced speech. During synthesis unvoiced excitation is often generated at the last vowels of the sentences. This produces a very unpleasant voice and it is not a proper model of glottalization. Fig. 3 a) and b) show an example for the end of a sentence synthesized by the baseline system showing the residual (a) and the final speech waveform (b). In the section denoted by a blue horizontal arrow unvoiced excitation was generated for some part of the vowel 'á', and therefore there is only aperiodic noise in the end of the speech signal.

3. HTS-CDBK extended with an irregular voice model

First, several acoustic properties of glottalization are introduced. Then an available semi-automatic regular-to-irregular transformation method is described. Finally, this method is further improved and integrated into HTS-CDBK. The novel system is denoted as HTS-CDBK+Irreg-Rule.

3.1. Acoustic properties of irregular phonation

In natural speech, irregular phonation can be distinguished from regular phonation by several properties ([13], [20]):

- the overall intensity level is lower
- the time that is elapsed between successive glottal pulses is longer and more irregular, resulting in lower F0 and higher jitter
- abrupt changes occur in the amplitude of the periods
- the open quotient (proportion of the glottal cycle where the glottis is open) is lower
- first formant bandwidth is increased because of more acoustic losses at the glottis
- the closure of the vocal folds is more abrupt

Some of these properties are observable in both the speech and in the residual signal. An example for this can be seen in Fig. 1. In the irregularly phonated interval the pitch is lower and the periods have clearly abrupt changes in amplitude.

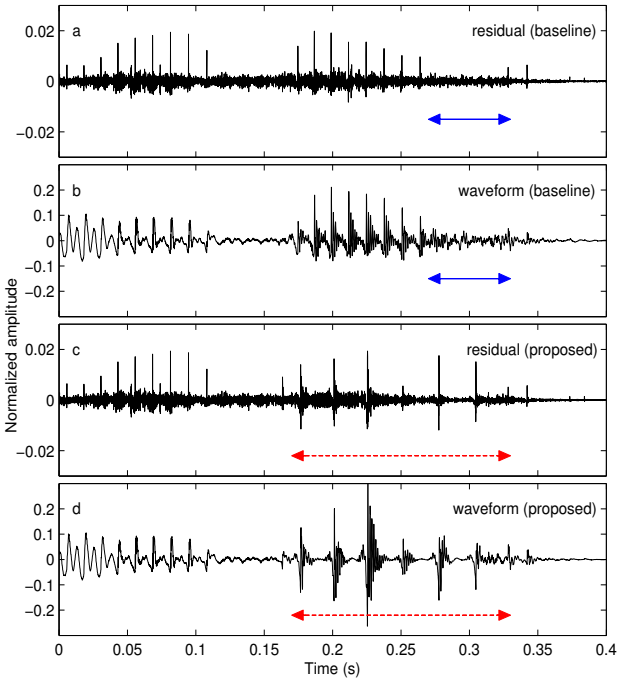


Figure 3: *Synthesized version of the word 'mihái' extracted from the end of a longer sentence with a) and b) from the baseline system and c) and d) from the proposed system.*

3.2. Regular to irregular transformation method

In [13], a regular-to-irregular voice transformation method was proposed which uses amplitude scaling of individual glottal cycles. Here, the modal speech is pitch-synchronously windowed, the periods are multiplied by individual hand-selected scaling factors and finally speech is overlap-added from the modified signal. The scaling factors can either boost, attenuate, remove or leave unmodified the cycles. [13] extends this with stylized pulse pattern copying yielding in a semi-automatic transformation method.

In the present form, this method is not suitable to be integrated into HTS; partly because it is manual or semi-automatic and as it works on the speech signal itself and not on excitation. However, the concepts of this transformation method were re-used and further improved yielding in an automatic model that was integrated into HTS-CDBK.

3.3. The proposed model

The proposed model differs from the baseline only in the synthesis phase. The analysis, training and the training speech

database are the same as in the baseline system (see Sections 2.1, 2.2 and 2.4, respectively).

The proposed model applies three heuristics similarly to [13]: 1) pitch halving, 2) pitch-synchronous residual modulation with periods multiplied by random amplitudes and 3) spectral distortion. Although the theoretical correctness of these heuristics cannot be proven because irregular phonation does not have a strict definition and each occurrence is different, in our preliminary experiments these ideas were useful and improved the baseline system. All of the heuristics are motivated by acoustic properties of irregular phonation, which are described in detail here:

1) In the sections that should be synthesized with irregular phonation, the half of the F0 of the generated parameter sequence is used. If there is F0=0 in the parameters of the glottalized section as in the baseline system, than before the halving the F0 is first interpolated according the neighboring frames. We applied the pitch halving because glottalization has often significantly lower F0 than modal speech (see Section 3.1), and [13] argues that by removing every second or third cycle the perception of samples is similar to decreasing the open quotient. In the residual codebook, frames with extremely low F0 are rare. Therefore, during synthesis, residual frames are zero padded which results in a similar effect than removing every second cycle.

2) During residual synthesis, each pitch cycle is multiplied by a random scaling factor in the range of {0..1}. This is similar to [13] but we do not boost any of the periods, only attenuate or leave them unchanged. This heuristic is motivated by the property of glottalization that is visible in Fig. 1: irregular phonation has often strong amplitude attenuations during the consecutive pitch cycles. From the modified residual periods the residual signal is obtained by overlap-adding the frames.

3) Finally, spectral distortion is applied. In [28] we found that the frame-by-frame MGC parameters of irregularly phonated speech are less smooth than those of regular speech. Therefore here we try to 'distort' the MGC parameters similarly by slightly modifying them: the parameter values are multiplied by random numbers between {0.995…1.005}. This yields a less smooth parameter sequence for each dimension of MGC. Note that one might argue that by adding random numbers to the residual or waveform samples itself the speech signal could be directly distorted. However, there is only a small chance that such a distortion would lead to a speech signal that is similar to original irregular utterances.

As there is no explicit glottalization model (e.g. irregular phonation labels, questions for decision trees) in the HTS-CDBK system, sections with irregular phonation have to be found from the generated F0 sequence. In our experiments the generated parameter and label files were checked automatically. Glottalization was applied if at least five consecutive frames were given zero F0 within a vowel. In these cases, fundamental frequency was interpolated between the voiced parts to have a straight F0 line, or was set to slightly decreasing if there were no voiced neighboring sounds.

Fig. 3 shows an example for the results of the baseline (HTS-CDBK: a, b) and the extended systems (HTS-CDBK+Irreg-Rule: c, d). In a) and b) the blue horizontal arrow shows the section where the excitation is unvoiced within the vowel 'á' in HTS-CDBK. As this section is longer than five frame shifts (25 ms), we apply glottalization for this vowel in the HTS-CDBK+Irreg-Rule system. In c) and d) the proposed residual and speech signal are shown and red dashed horizontal line indicates the glottalized vowel 'á'. It is clearly visible on both the residual and speech signals that the extended model is closer to the original irregular signal (Fig. 1) than the baseline system.

4. Perceptual evaluation

In order to evaluate the quality that can be achieved by the proposed HTS-CDBK+Irreg-Rule method, a listening test was conducted according to the guidelines of [29]. A major factor that determines the usefulness of this method is if human listeners accept the synthesized speech. Therefore, our aim was to measure the perceived 'pleasantness' and the similarity to the original speaker. Synthesized samples of the baseline system were compared to those of the proposed solution.

4.1. Methods

To create the speech stimuli, four voice models with the two systems and the two speakers were created. Note that HTS-CDBK and HTS-CDBK+Irreg-Rule only differ in the synthesis part, therefore the analysis, training and speech data was the same here. 130-130 sentences were synthesized with all four voice models and 10-10 sentences having at least one irregularly synthesized vowel at the end were selected for the subjective test. The last word (containing at least two syllables) of each sentence was cut and used as stimuli as we wanted the listeners to focus only on the sentence endings. An example for an utterance from the test can be seen in Fig. 3.

In the test, the two versions of each word were included, resulting altogether 40 utterances (2 speakers · 10 words · 2 versions). A web-based paired comparison test with two CMOS-like questions was created. Before the test, listeners were asked to listen to an example from speaker FF3. In the first part of the test, the listeners had to rate their preference ('Which version do you think is more pleasant?', '1 – first is much more pleasant' … '5 – second is much more pleasant'). In the second part, they were asked which version is more similar to the original speaker. For this, a reference speech sample was shown first and the two stimuli after that ('Which version is more similar to the original speaker?', '1 – first is more similar', '2 – equal', '3 – second is more similar'). The utterances were presented in a randomized order.

4.2. Results

Altogether 11 listeners participated in the test. They were all university students or computer science professionals, between ages of 19-30 years. All of them were native speakers of Hungarian and none of them reported any hearing loss. On the average the whole test took 9 minutes to complete.

The results of the listening test are presented in Fig. 4 for the two speakers. The figure provides a comparison between the baseline system (left part, blue color) and the proposed system (right part, red color). It can be seen that for the preference question, for both speakers the results are higher than the equal answer of 50% (CMOS score=3.0) meaning that the proposed system was more preferred (mean altogether: 3.36). Similarity scores are higher than the equal 50% (CMOS=2.0) for both speakers FF3 and FF4 (mean altogether: 2.38). The ratings of the listeners were compared by t-tests as well. The statistical analysis showed that the proposed method was significantly preferred in terms of 'pleasantness' ($p<0.0005$) and was significantly more similar to the original speaker ($p<0.0005$) than the baseline system. By investigating

the scores for the stimuli one by one, we found that all of the utterances ranked higher in the similarity test, while in 18 out of 20 sample pairs the extended model was preferred.

From this subjective experiment, we can conclude that the HTS-CDBK+Irreg-Rule system improves the perceived naturalness of synthesized speech using an irregular voice model and the proposed method can generate speech that is more similar to the original speaker.

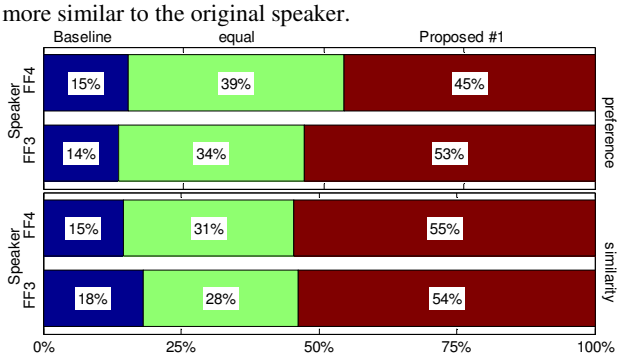


Figure 4: *Results of the subjective evaluation showing percentages of Comparative MOS scores between baseline and proposed systems.*

5. Acoustic evaluation

The perception test showed the preference of the proposed model. However, from the listening test results it is not known whether the proposed system models irregular voice properly or it was just preferred to use other excitation instead of white noise in the investigated vowels. Therefore we investigated several acoustic cues which were found previously to distinguish original irregular and regular speech [13].

5.1. Methods

The acoustic properties of glottalization were introduced in Section 3.1. In the acoustic experiment the three most important acoustic cues [20] are used: open quotient (OQ), first formant bandwidth (B1) and spectral tilt (TL). OQ and TL are expected to be lower for irregular phonation, while B1 is increased compared to regular voice. If the synthesized utterances match these correlates, that might provide an explanation for their perceptual acceptability.

The above parameters are more convenient to consider in the frequency domain; therefore the changes in H1-H2 (the difference of the amplitudes of the first two harmonics), H1-A1 (H1 relative to the first formant amplitude) and H1-A3 (H1 relative to the third formant amplitude) were measured which are correlated with OQ, B1 and TL, respectively [31, 32]. These parameter values can be biased by the effects of the first three formants. To compensate this, we used the equations suggested by [30] and implemented in VoiceSauce: the value of H1 and H2 was corrected for F1 and F2 (H1* and H2*), and the value of A3 was corrected for F1, F2, and F3 (A3*).

The measurements were conducted partly on the stimuli used in the perceptual evaluation (10-10 words synthesized by the proposed model). The other part of the investigated speech material consisted of 10-10 original regular and original irregular vowels selected from the PPBA database from both speakers. Altogether the parameters of 80 vowels were measured. First the wave files were resampled to 8 kHz. Then a glottalized vowel from the original irregular version was selected and the middle of the vowel (roughly aligned with the pitch marks) was chosen and the same vowel was measured in the original regular version. In the synthesized versions, the vowels created by the irregular voice models were measured. In Wavesurfer, the 512-point FFT spectrum, calculated using a Hamming window, was displayed at the chosen locations and the parameters were graphically measured. In the irregular versions often strong subharmonics appeared; here we measured H1 and H2 as the lowest two of the spectral peaks.

5.2. Results

The mean values of H1*-H2* (proportional to OQ), H1*-A1 (proportional to 1/B1) and H1*-A3* (proportional to TL) are shown in Fig. 5 for the three utterance versions separately. In one-way ANOVAs, stimulus type had a significant effect on the difference between the first two harmonics ($p<0.0005$), while the other two calculated parameters were not significantly different. Tukey's post hoc test was used to compare the mean parameter values of each stimulus type.

H1*-H2* was significantly different for the original regular speech ($p<0.05$) whereas it was approximately the same for the original irregular and for the synthesized irregular recordings ($p=0.97$, n.s.). This means that in terms of open quotient, the synthesized versions are close to the original irregular versions. H1*-A1 and H1*-A3* are not significantly different for any of the groups, but in the figure we can see the trends that the irregular voice model have created. In terms of the H1*-A1 and first formant bandwidth the synthesized irregular utterances are close to the original irregular recordings. In this experiment, H1*-A3* was not helpful to differentiate between the regular and irregular utterances.

From the acoustic experiment the conclusion is that the proposed irregular model can reconstruct two of the three investigated acoustical correlates of irregular speech.

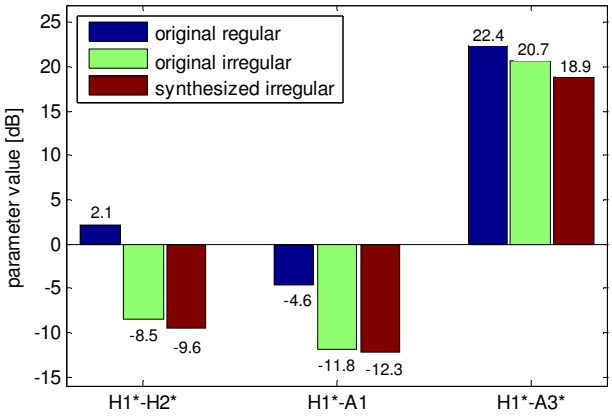


Figure 5: *Results of the acoustic experiment.*

6. Discussion and Conclusions

This paper presented a method to synthesize irregular voice within the HTS framework. The proposed method uses pitch halving, amplitude scaling of the pitch periods of the residual signal and spectral distortion. Although the theoretical correctness of these heuristics cannot not be proven because irregular phonation does not have a strict definition and every occurrence is different, in our experiments these ideas were useful and improved the baseline system. The proposed method was supported by perception and acoustic tests. A perception experiment found the proposed method to synthesize glottalized speech that is closer to the original speaker while increasing naturalness. An acoustic experiment

found the output of the model to be similar to original irregular speech in terms of open quotient and first formant bandwidth.

The new method is fully automatic because amplitude scales are determined randomly and no manual scaling is necessary. By applying predefined stylized pulse patterns as in [13] instead of random scaling factors, the naturalness of synthesized glottalization might be further improved. With the application of an irregular vs. regular classification algorithm (e.g. [14]), glottalization could be modeled explicitly in HTS. To create a full speech synthesis system that is able to synthesize irregular speech, it will be necessary to include new contextual factors or additional parameter streams like in [17]. In [33] we extend this model and show another data-driven approach for irregular voice synthesis.

With the new method we extend previous speech processing techniques dealing with irregular phonation: it may contribute to building natural, emotional and personalized speech synthesis. Irregular phonation is frequently adopted in lively story-telling, natural interactive conversation [17] and can signal sadness [18] or boredom [19]. Therefore an irregular voice model improves expressive speech synthesis systems. For example it is possible to create speaker adaptation for deep elderly voices (e.g. those of famous radio announcers) having frequent glottalization.

7. Acknowledgements

We would like to thank the listeners for participating in the subjective test. This research was partially supported by the Paelife (Grant No AAL-08-1-2011-0001), the EITKIC_12-1-2012-001 and the CESAR (Grant No 271022) projects.

8. References

[1] H. Zen, T. Nose, J. Yamagishi, S. Sako, T. Masuko, and A. W. Black, "The HMM-based speech synthesis system version 2.0," in Proc. ISCA SSW6, 2007, pp. 294–299.

[2] H. Zen, T. Toda, M. Nakamura, and K. Tokuda, "Details of the Nitech HMM-based speech synthesis system for the Blizzard Challenge 2005," IEICE Transactions on Information and Systems, vol. E90-D, no. 1, pp. 325–333, 2007.

[3] G. Fant, J. Liljencrants, and Q. Lin, "A four-parameter model of glottal flow," STL-QPSR, vol. 4, pp. 1–13, 1985.

[4] J. Cabral, S. Renals, K. Richmond, and J. Yamagishi, "Towards an Improved Modeling of the Glottal Source in Statistical Parametric Speech Synthesis," in Proc. ISCA SSW6, 2007, pp. 113–118.

[5] T. Drugman, G. Wilfart, and T. Dutoit, "A deterministic plus stochastic model of the residual signal for improved parametric speech synthesis," in Proc. Interspeech, 2009, pp. 1779–1782.

[6] T. Raitio, A. Suni, H. Pulakka, M. Vainio, and P. Alku, "HMM-based Finnish text-to-speech system utilizing glottal inverse filtering," in Proc. Interspeech, 2008, pp. 1881–1884.

[7] A. Suni, T. Raitio, M. Vainio, and P. Alku, "The GlottHMM Entry for Blizzard Challenge 2012: Hybrid Approach," in Blizzard Challenge 2012, 2012.

[8] Z. Wen and J. Tao, "Inverse Filtering Based Harmonic plus Noise Excitation Model for HMM-based Speech Synthesis," in Proc. Interspeech, 2011, pp. 1805–1808.

[9] J. S. Sung, D. H. Hong, H. W. Koo, and N. S. Kim, "Statistical Approaches to Excitation Modeling in HMM-Based Speech Synthesis," IEICE Transactions on Information and Systems, vol. E96-D, no. 2, pp. 379–382, 2013.

[10] T. G. Csapó and G. Németh, "A novel codebook-based excitation model for use in speech synthesis," in IEEE CogInfoCom, 2012, pp. 661–665.

[11] T. G. Csapó and G. Németh, "Statistical parametric speech synthesis with a novel codebook-based excitation model," Intelligent Decision Technologies, accepted, 2013.

[12] M. Blomgren, Y. Chen, M. L. Ng, and H. R. Gilbert, "Acoustic, aerodynamic, physiologic, and perceptual properties of modal and vocal fry registers," JASA, vol. 103, pp. 2649–2658, 1998.

[13] T. Bőhm, N. Audibert, S. Shattuck-Hufnagel, G. Németh, and V. Aubergé, "Transforming modal voice into irregular voice by amplitude scaling of individual glottal cycles," in Acoustics'08, 2008, pp. 6141–6146.

[14] T. Bőhm, Z. Both, and G. Németh, "Automatic Classification of Regular vs. Irregular Phonation Types," in NOLISP, 2009, pp. 43–50.

[15] L. Dilley, S. Shattuck-Hufnagel, and M. Ostendorf, "Glottalization of word-initial vowels as a function of prosodic structure," JPhon, vol. 24, no. 4, pp. 423–444, Oct. 1996.

[16] C. Gobl and A. N. Chasaide, "The role of voice quality in communicating emotion, mood and attitude," Speech Communication, vol. 40, no. 1–2, pp. 189–212, Apr. 2003.

[17] T. Drugman, J. Kane, T. Raitio, and C. Gobl, "Prediction of Creaky Voice from Contextual Factors," in Proc. ICASSP, 2013.

[18] Cs. Zainkó, M. Fék, and G. Németh, "Expressive Speech Synthesis Using Emotion-Specific Speech Inventories," Lecture Notes in Computer Science, no. 5042, pp. 225–234, 2008.

[19] J. Laver, The Phonetic Description of Voice Quality. Cambridge: Cambridge University Press, 1980.

[20] D. H. Klatt and L. C. Klatt, "Analysis, synthesis, and perception of voice quality variations among female and male talkers.," JASA, vol. 87, no. 2, pp. 820–857, 1990.

[21] A. V. McCree and T. P. Barnwell, "A mixed excitation LPC vocoder model for low bit rate speech coding," IEEE Trans. Speech and Audio Processing, vol. 3, no. 4, pp. 242–250, 1995.

[22] H. Silén, E. Helander, J. Nurminen, and M. Gabbouj, "Parameterization of vocal fry in HMM-based speech synthesis," in Proc. Interspeech, 2009, pp. 1775–1778.

[23] T. Drugman, J. Kane, and C. Gobl, "Modeling the Creaky Excitation for Parametric Speech Synthesis," in Proc. Interspeech, 2012, pp. 1424–1427.

[24] T. Drugman and M. Thomas, "Detection of glottal closure instants from speech signals: a quantitative review," IEEE Transactions on Audio, Speech and Language Processing, vol. 20, no. 3, pp. 994–1006, 2012.

[25] G. de Krom, "A cepstrum-based technique for determining a harmonics-to-noise ratio in speech signals," Journal of Speech and Hearing Research, vol. 36, no. 2, pp. 254–266, Apr. 1993.

[26] A. J. Hunt and A. W. Black, "Unit selection in a concatenative speech synthesis system using a large speech database," in Proc. ICASSP, 1996, vol. 1, pp. 373–376.

[27] G. Olaszy, "Precíziós, párhuzamos magyar beszédadatbázis fejlesztése és szolgáltatásai [Development and services of a Hungarian precisely labeled and segmented, parallel speech database] (in Hungarian)," Beszédkutatás 2013 [Speech Research 2013], 2013.

[28] T. G. Csapó and G. Németh, "Transformation of irregular voice to regular voice by residual analysis and synthesis," IEEE Signal Processing Letters, 2013, in preparation.

[29] R. A. J. Clark, M. Podsiadlo, M. Fraser, C. Mayo, and S. King, "Statistical Analysis of the Blizzard Challenge 2007 Listening Test Results," in Blizzard Challenge 2007, 2007.

[30] M. Iseli and A. Alwan, "An improved correction formula for the estimation of harmonic magnitudes and its application to open quotient estimation," in Proc. ICASSP 2004, pp. 669–672.

[31] E. B. Holmberg, R. E. Hillman, J. S. Perkell, P. C. Guiod, and S. L. Goldman, "Comparisons among aerodynamic, electroglottographic, and acoustic spectral measures of female voice," J.Speech Hear.Res, vol. 38, no. 6, pp. 1212–1223, 1995.

[32] N. Henrich, C. d'Alessandro, B. Doval, "Spectral correlates of voice open quotient and glottal flow asymmetry: theory, limits and experimental data", Proc. Eurospeech 2001, pp. 47–50.

[33] T. G. Csapó, G. Németh, „Modeling irregular voice in statistical parametric speech synthesis with residual codebook based excitation," IEEE Journal on Selected Topics in Signal Processing, submitted, 2013.

Expression of Speaker's Intentions through Sentence-Final Particle/Intonation Combinations in Japanese Conversational Speech Synthesis

Kazuhiko Iwata, Tetsunori Kobayashi

Perceptual Computing Laboratory, Waseda University, Japan

Abstract

Aiming to provide the synthetic speech with the ability to express speaker's intentions and subtle nuances, we investigated the relationship between the speaker's intentions that the listener perceived and sentence-final particle/intonation combinations in Japanese conversational speech. First, we classified F0 contours of sentence-final syllables in actual speech and found various distinctive contours, namely, not only simple rising and falling ones but also rise-and-fall and fall-and-rise ones. Next, we conducted subjective evaluations to clarify what kind of intentions the listeners perceived depending on the sentence-final particle/intonation combinations. Results showed that adequate sentence-final particle/intonation combinations should be used to convey the intention to the listeners precisely. Whether the sentence was positive or negative also affected the listeners' perception. For example, a sentence-final particle *'yo'* with a falling intonation conveyed the intention of an "order" in a positive sentence but "blame" in a negative sentence. Furthermore, it was found that some specific nuances could be added to some major intentions by subtle differences in intonation. The different intentions and nuances could be conveyed just by controlling the sentence-final intonation in synthetic speech.

Index Terms: speech synthesis, speaker's intention, sentence-final particle, sentence-final intonation, conversational speech

1. Introduction

Speech synthesis technology has made remarkable progress in its synthetic voice quality, and the natural-sounding synthetic speech has recently become available. Moreover, various approaches to synthesizing expressive and conversational speech have been also reported in the past decade [1, 2, 3, 4]. However, due to the diversity of conversational speech, numerous problems still need to be solved to build a useful speech synthesis system for robots and speech-enabled agents that communicate with humans through synthetic speech. We communicate with each other through linguistic and paralinguistic information [5]. An utterance is affected by the speaker's intentions, attitudes, feelings, personal relationship with the listeners, and so forth. The paralinguistic features of the utterance vary a great deal.

In spoken Japanese, a speaker's intention is usually conveyed at the end of a sentence by sentence-final particles or auxiliary verbs [6]. The functions of sentence-final particles have been extensively studied in the field of linguistics [7, 8, 9, 10, 11, 12, 13]. For example, a sentence-final particle *'yo'* indicates a strong assertion, and *'ne'* indicates a request for listeners' agreement. In addition, the intonation of the sentence-final particle, namely, the sentence-final intonation plays a significant role in expressing the intention and has also been studied over the years [14, 15, 16, 17, 18, 19]. For instance, a sentence with a sentence-final particle *'ka'* becomes a declarative sentence with a falling intonation, whereas it becomes an interrogative sentence with a rising intonation. Moreover, it can express various additional nuances such as surprise, admiration, concern, doubt, or anger through different intonations. The functions of the sentence-final and phrase-final intonation have been discussed also in terms of turn-taking [20, 21]. However, not only the sentence-final intonation but also the intonation of the whole sentence varies depending on the speaker's intention or attitude. In some languages other than Japanese, it has been reported that the listeners could identify the speaker's attitudes before the end of the sentences [22, 23]. In contrast, the experiment that used two utterances of the same sentence uttered with different intentions demonstrated the importance of the sentence-final intonation in Japanese. When the speech segments of the sentence-final particle were cut off and swapped, the listeners perceived the intention that was expressed by the segment of sentence-final particle though the overall F0 contours of the utterances differed from each other [24].

On the other hand, there are not as yet many approaches to expressing the speaker's intention by controlling the sentence-final intonation in the field of speech synthesis technology. Boundary pitch movements in Tokyo Japanese have been analyzed and modeled [25]. Five boundary pitch movements were chosen and their meanings were examined through association with eight semantic scales. A prosody control method has been proposed [26], which was based on the analysis of intonations of the one-word utterance *'n'*. This study revealed that speaking attitudes were expressed by the F0 average height and dynamic pattern of the word *'n'*, since it has no particular lexical meaning. However, none of these models above considered the association of the expressions of the speaker's intention with the sentence-final particles despite the fact that the intention is verbally expressed by the sentence-final particle in Japanese spoken dialogue. Although we investigated the relationship between the speaker's intentions and sentence-final intonation contours in a previous study [27], we did not consider the relevance of the sentence-final particles.

In this study, we focus on the listeners' perception of speaker's intention, associating with sentence-final particles and their intonation contours in order to enable synthetic speech to express various intentions. In Section 2, we classify sentence-final intonation contours to find what kinds of sentence-final intonation were used in actual speech. In Section 3, we select several distinctive intonation contours from among the classification results and conduct a subjective evaluation to find suitable sentence-final particle/intonation combinations for conveying some specific intentions (hereafter "major intentions").

Furthermore, in Section 4, we investigate sentence-final particle/intonation combinations that can add subtle nuances to the major intentions aiming to provide the synthetic speech with wide expressiveness. In Section 5, we conclude the paper with some remarks.

2. Classification of sentence-final intonation

2.1. Previous research in linguistics

As mentioned above, the functions of the sentence-final intonation have been discussed and various views have been proposed by the linguists. Table 1 shows some examples for the categorization of the sentence-final intonation in terms of the contours. In the majority of them, the sentence-final intonation contours were classified into two categories or up to five categories at most. However, there seems to be no accepted notion.

Table 1: *Examples for categorization of sentence-final intonation in linguistics.*

Number of categories	Categories
2	Rise, Fall [15]
4	Rise, Fall-and-rise, Fall, Rise-and-fall [16]
5	Interrogative rise, Prominent rise, Fall, Rise-and-fall, Flat [17]

2.2. Speech data

We used speech data that were created with the aim of developing an HMM-based speech synthesis system that had multiple HMMs depending on situations of conversation [4]. To build the HMMs, we designed several situations and more than 2000 sentences derived from dialogues that our communication robot [28] performed. These sentences were uttered by a voice actress, to whom we did not indicated any specific intentions for each sentence but explained the situations in which the robot was supposed to utter each sentence. The F0 contours were extracted using STRAIGHT [29], and the phonemes were manually segmented. The intonation contours at the end of these utterances varied a great deal and expressed subtle nuances and connotations. Of these data, 2092 utterances whose sentence-final vowel was not devoiced were used for the analysis.

The sentence-final intonation contours, that is, the F0 contour in the sentence-final syllable was extracted by referring to the phoneme boundaries. Because the actual F0 values of the utterances differed from each other and were difficult to classify, the time and frequency axes were normalized. To remove F0 perturbation caused by jitter and microprosody, the logarithmic F0 contour was approximated by a third-order least squares curve. The approximated curve was sampled at 11 points that equally divided the duration into 10. Finally, the starting point of the sampled curve was parallel-translated to the origin [27].

2.3. Classification of sentence-final intonation contours

The normalized F0 contours obtained by the above process were classified by Ward's clustering algorithm [30], which is one of the hierarchical clustering algorithms and merges clusters so as to minimize the within-cluster variance.

Figure 1 shows an example of the clustering results when the number of clusters was set to 32. The F0 contours denoted by thick circles and a thick line are the centroids of each cluster. The numbers in square brackets (e.g., [C2]) are expedient cluster IDs corresponding to the clustering sequence. Note that the lengths of the vertical lines do not represent the distance between the clusters due to the limitation of the page layout. Various sentence-final intonation contours were found, including not only simple rising and falling intonations but also rise-and-fall and fall-and-rise intonations.

2.4. Perceptual discrimination of intonation contours by centroids

We found the sentence-final intonation contours were classified into distinctive clusters. However, we predicted not all the pairs of cluster centroids would have a notable perceptual difference from each other because the clustering was based only on the shapes of the F0 contours. Therefore, a preliminary evaluation was conducted.

A back-channel utterance *'haa'*, which had no specific linguistic meaning, was resynthesized by the STRAIGHT vocoder [29], and its F0 contour was replaced with 127 centroid pairs obtained in the process of classifying the F0 contours into 128 clusters. Fifteen listeners were randomly presented with 254 *'haa'* pairs including a reverse order for each pair of centroids and then asked whether they perceived the two intonations to be the same or different. The results of the evaluation are shown in Figure 2. The numbers in parentheses indicate the number of responses when the intonations by the centroids on both sides were perceived as different. This is how we obtained the criteria for sifting through and selecting the F0 contours that would be used in the next experiments.

3. Major intentions conveyed by sentence-final particle/intonation combinations

3.1. Selection of representative intonation contours

We consulted previous studies prior to conducting the subjective evaluation to investigate what kind of speaker's intentions could be conveyed by sentence-final particles and their intonation contours. Referring to the results of the preliminary evaluation (Figure 2), we stopped dividing clusters whose child clusters received 28 or fewer perceptions that their intonations were different. The selected clusters were C5, C20, C6, C15, C16, and C19.

Compared with the categorization in the previous research shown in Table 1, the centroid of the cluster C5 seems to correspond to the interrogative rise intonation, C20 to the fall-and-rise, C6 to the fall, C15 to the rise-and-fall, C16 to the prominent rise, and C19 to the flat. These results indicate that specific intonation contours corresponding to the categories in the previous research could be obtained from the speech database.

3.2. Experimental setup

A subjective evaluation was conducted to clarify what kind of intentions the listener could perceive through the sentence-final intonations produced by the selected six centroids.

We prepared 31 short sentences consisting of a verb *'taberu'* ("eat") followed by a sentence-final particle (*'yo'*, *'na'*, *'ne'*, *'datte'*, etc.), an auxiliary verb (*'daroo'*), or one of their concatenations (*'yone'*, *'yona'*, *'datteyo'*, *'daroone'*, etc.). Synthetic voices of these sentences were generated by our HMM-based speech synthesis system [4]. The duration of the last vowel of each sentence was fixed to 313 ms, which was the mean duration of the last vowels of the sentences in the speech

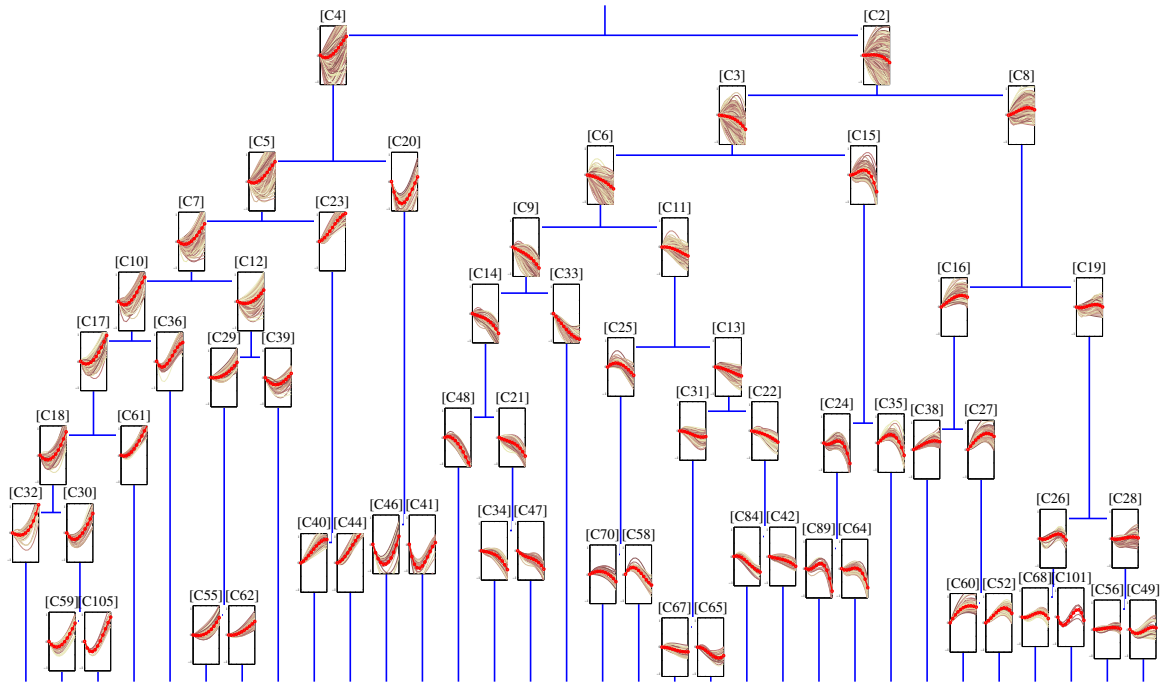


Figure 1: *Result of clustering sentence-final intonation contours when the number of clusters was set to 32.*

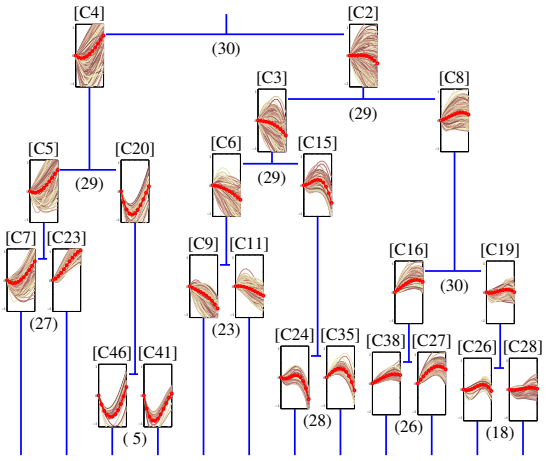


Figure 2: *Preliminary evaluation results of intonations generated by cluster centroids. The numbers in parentheses indicate the number of responses when the intonations by the centroids on both sides were perceived as different.*

data. Then, the sentence-final F0 contour was replaced with the six centroids. We also designed 11 speaker's intentions ("request", "order", "blame", "hearsay", "guess", "question", etc.) and situations where these intentions could be indicated. We informed 20 listeners of the situations and speaker's intentions and asked them to evaluate whether or not both the lexical and intonational expressions of the stimulus were suitable for conveying the intention on a five-level scale: –2 (unsuitable; suitable for a different intention), –1 (rather unsuitable), +1 (rather suitable), +2 (suitable), and 0 (none of the above).

3.3. Results and discussion

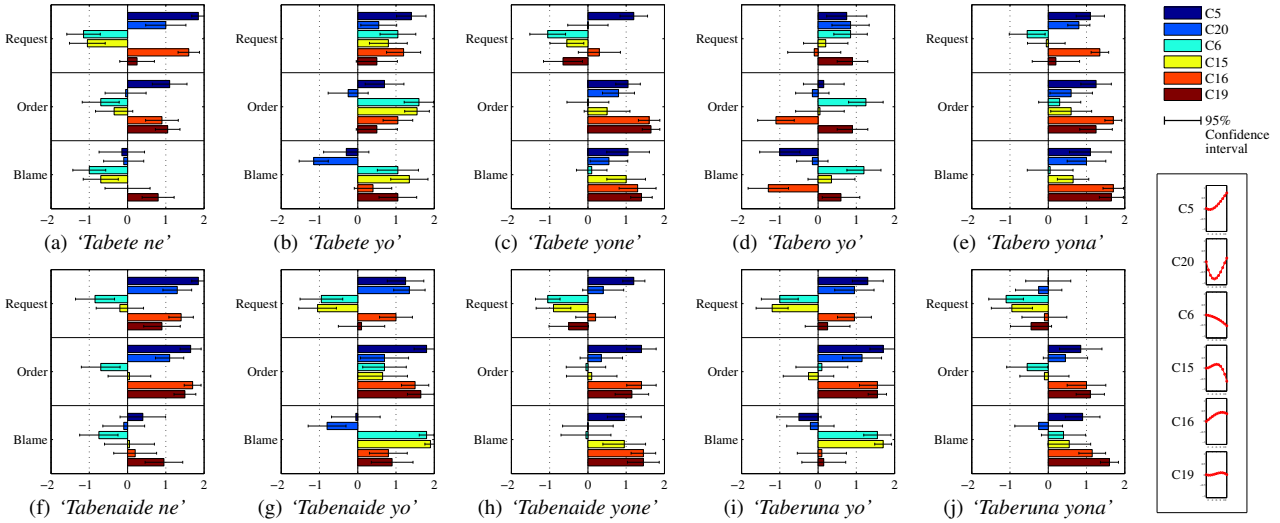
Figure 3 shows the key results of the subjective evaluation, with a particular focus on a "request", an "order", and "blame".

- Sentence-final particle *'ne'*
  Generally, the use of *'ne'* signals a polite "request". This was endorsed with the rising intonations C5 and C20 (Figures 3(a) and 3(f)). In the positive sentence *'Tabete ne'*, a "request" was also conveyed with the rising C16 intonation. On the other hand, in the negative sentence *'Tabenaide ne'* with the rising C16 and flat C19 intonations, an "order" was perceived more clearly than in the positive sentence.

- Sentence-final particle *'yo'*
  In the positive sentences *'Tabete yo'* (Figure 3(b)) with the falling intonations C6 and C15 and *'Tabero yo'* (Figure 3(d)) with C6, an "order" (*"Don't eat."*) was conveyed. In contrast, in the negative sentences *'Tabenaide yo'* (Figure 3(g)) and *'Taberuna yo'* (Figure 3(i)), which meant prohibition, "blame" (*"Why did you eat even though I told you not to?"*) was strongly conveyed with the falling intonations C6 and C15. When with the rising and flat intonations C5, C16, and C19, they caused an "order" impression.

- Sentence-final particles *'yone'* and *'yona'*
  'Yone' is known to have different lexical functions from *'ne'* and *'yo'*. "Blame", which was not much perceived in the sentences with *'ne'*, was conveyed with the rising C16 and flat C19 intonations (Figures 3(c) and 3(h)). This tendency differs from the case with *'yo'*, where "blame" was conveyed with the falling intonations C6 and C15. "Blame" could be conveyed also in *'yona'* with C16 and C19 (Figures 3(e) and 3(j)).

To summarize the results, we can obtain Table 2, which shows the highest scored intentions for each combination. Then we can consult this table when we generate synthetic speech. For example, we can express "request" in a positive sentence with a sentence-final particle *'ne'* by using C5 or C20 as the sentence-final intonation. When we need to express "blame" in a negative sentence with a sentence-final particle *'yo'*, we should use C6 or C15.



(a) *'Tabete ne'* (b) *'Tabete yo'* (c) *'Tabete yone'* (d) *'Tabero yo'* (e) *'Tabero yona'*

(f) *'Tabenaide ne'* (g) *'Tabenaide yo'* (h) *'Tabenaide yone'* (i) *'Taberuna yo'* (j) *'Taberuna yona'*

Figure 3: *Subjective evaluation results of major intentions depending on sentence-final particle/intonation combinations. The sentences in (a), (b), (c), (d), and (e) are positive (roughly, "Please eat."), and the others are negative ("Please don't eat.").*

Table 2: *Speaker's intention conveyed by sentence-final particle/intonation combination. The intentions ("request", "order", and "blame") that received the highest positive score are shown by the initial letters and underlined when their scores are higher than or equal to 1.0 (rather suitable). *** $p<0.001$, ** $p<0.01$, * $p<0.05$, + $p<0.1$, where p is the maximum p-value among two comparisons.*

Sentence-final particle (Phrase)	Sentence-final intonation C5	C20	C6	C15	C16	C19
'Tabete ne'	**R*****	**R****	–	–	**R***	**O**
'Tabenaide ne'	**R**	**R**	–	O/B	**O***	**O***
'Tabete yo'	**R****	R**	**O**+	**O**	**R**	**B**
'Tabenaide yo'	**O**+	**R***	**B****	**B*****	**O***	**O***
'Tabete yone'	**R**	O	B	**B**	**O**	**O**
'Tabenaide yone'	**O**	R	–	B**	**B**	**B**
'Tabero yo'	R	R**	**O**	B	–	R/O
'Taberuna yo'	**O**	**O**	**B*****	**B*****	**O**+	**O*****
'Tabero yona'	**O**	**B**	O	B	**O/B**	**B**
'Taberuna yona'	B	O*	B**	B	**B**	**B***

4. Additional nuances conveyed by sentence-final particle/intonation combinations

As the next step of this study, we investigated whether the listeners could perceive additional intentions, attitudes, or feelings (hereinafter collectively called "additional nuances") through the sentence-final intonations.

4.1. Selection of representative intonation contours

We increased the types of representative intonation contours to be used for the subjective evaluation of additional nuances. The selected six clusters in **3.1** were divided into several subclusters. This time, we merged two clusters from the 128 leaf nodes to the six clusters. When two clusters received 27 or more out of 30 (more than or equal to 90%) perceptions that their intonations were different, they were not merged. Additionally, their parent cluster was not merged with its paired cluster either. Thus, the cluster C5 was ultimately divided into 8 subclusters, C20 into 3, C6 into 11, C15 into 5, C16 into 7, and C19 into 7, as listed in Table 3.

4.2. Experimental setup

We defined the intentions of a "request", an "order", and "blame" as the major intentions for this evaluation. Considering the results of the previous section (Table 2), we chose sentence-final particle/intonation combinations from among those which received the significantly different ($p < 0.05$) score: *'Tabete ne'* with the sentence-final intonations by the subclusters of C5, C20, and C16 for a "request"; *'Tabenaide yo'* with the subclusters of C16 and C19 for an "order"; and *'Tabenaide yo'* with the subclusters of C6 and C15 for "blame". We generated 18 synthetic utterances with different intonations for the major intention "request", 14 utterances for the "order", and 16 utterances for the "blame" as the stimuli.

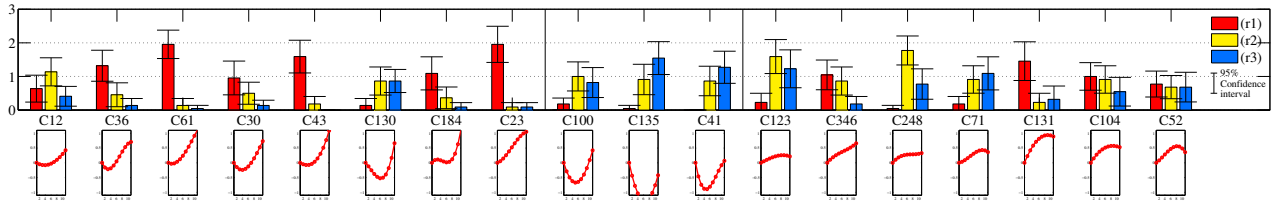
Three additional nuances for each major intention (Table 4) were designed, which one of the authors perceived from these stimuli in advance. The stimuli were presented to 22 listeners,

Table 3: *Selected subclusters as representative intonation contours to be used for the evaluation of additional nuances. Their centroids are shown in Figure 4.*

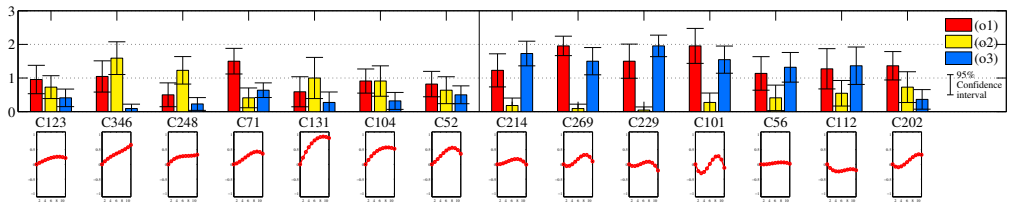
Parent cluster	Subclusters #	IDs
C5	8	C12, C36, C61, C30, C43, C130, C184, C23
C20	3	C100, C135, C41
C6	11	C9, C70, C334, C282, C232, C139, C151, C233, C125, C65, C22
C15	5	C24, C329, C117, C74, C155
C16	7	C123, C346, C248, C71, C131, C104, C52
C19	7	C214, C269, C229, C101, C56, C112, C202

Table 4: *Evaluated additional nuances for each major intention.*

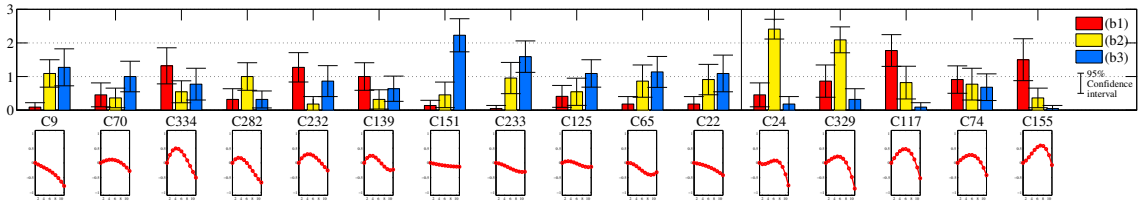
Major intention	Additional nuance The speaker seems	
Request	(r1)	to sincerely request the listener to eat.
	(r2)	not to really want the listener to eat.
	(r3)	to be slightly in a bad mood.
Order	(o1)	to be afraid the listener will definitely eat.
	(o2)	not to be afraid the listener will definitely eat.
	(o3)	to be in a slightly bad mood.
Blame	(b1)	to be in a hurry to stop the listener from eating.
	(b2)	to be in a rage after the listener has eaten.
	(b3)	to be disheartened after the listener has eaten.



(a) *'Tabete ne'* with major intention "request".



(b) *'Tabenaide yo'* with major intention "order".



(c) *'Tabenaide yo'* with major intention "blame".

Figure 4: *Subjective evaluation results of additional nuances to major intentions. The centroids of the subclusters of C5, C20, and C16 were used for expressing the major intention "request", those of C16 and C19 for "order", those of C6 and C15 for "blame".*

along with the situation of the dialogue and the additional nuances for each of the three major intentions. The listeners were asked to evaluate whether they felt the additional nuances on a four-level scale: 3 (strongly felt), 2 (felt), 1 (slightly felt), and 0 (not felt). In addition, they were asked to freely describe any other nuances they felt and evaluate the stimuli in the same way.

4.3. Results and discussion

The results are shown in Figure 4. Several intonation contours were found to be able to convey some additional nuances.

- "Request" (Figure 4(a))
  The additional nuances (r1) and (r2) have mutually exclusive meanings. C61, C23, C43, and C131 conveyed the nuance (r1) well. They all rise toward a considerably high frequency at the end of a sentence, which is a characteristic that can be considered to convey "sincerity" or "cordiality". In contrast, C248 and C123, which were slightly rising and rather flat overall contours, implied (r2). C135 and C41, which were fall-and-rise contours with the extreme lowering, insinuated (r3). As for the free description, some listeners noted that C248 and C346 created a sense of familiarity and caused an impression that the speaker was just like a senior (such as a parent of a friend). However, whether others similarly perceive it or not needs to be investigated further.

- "Order" (Figure 4(b))
  The additional nuances (o1) and (o2) are mutually exclusive. C269, C101, and C229, which were undulated, conveyed (o1). C71, which were slightly rising and undulated, also conveyed (o1). On the other hand, (o2) was not very clearly conveyed, but by C248. C229, C214, and C101 conveyed (o3) in addition to (o1).

- "Blame" (Figure 4(c))
  The additional nuance (b1) seemed to be expressed by a large rise-and-fall movement such as C117 and C155. C24 and

C329, which had a steep fall after slight rise, expressed (b2) clearly. C151 and C233, which had a slight fall, expressed (b3).

5. Conclusions

We investigated the expression of speaker's intentions through sentence-final particle/intonation combinations in Japanese conversational speech. Results showed that the sentence-final intonation contours varied a great deal and that adequate sentence-final particle/intonation combinations should be used to convey the intention to the listeners precisely. Furthermore, it was found that some specific nuances could be added to some major intentions by subtle differences in intonation.

We indicated that not only major intentions but also subtle nuances could be expressed by sentence-final particle/intonation combinations. However, other prosodic features such as the duration, power, and voice quality of the sentence-final syllable must be changed in actual speech to contribute to conveying the intentions. They need to be adequately controlled in order to express the intentions more precisely and more expressively. We intend to elucidate the relationship between these features and the intentions in our future work.

6. Acknowledgements

This work was supported in part by Global COE Program "Global Robot Academia".

7. References

[1] E. Eide, A. Aaron, R. Bakis, W. Hamza, M. Picheny, and J. Pitrelli, "A corpus-based approach to expressive speech synthesis," in *Proc. 5th ISCA ITRW on Speech Synthesis*, 2004, pp. 79–84.

[2] Y. Sagisaka, T. Yamashita, and Y. Kokenawa, "Generation and perception of F0 markedness for communicative speech synthesis," *Speech Communication*, vol. 46, no. 3–4, pp. 376–384, July 2005.

[3] M. Schröder, "Expressive speech synthesis: Past, present, and possible futures," in *Affective Information Processing*, J. Tao and T. Tan, Eds. London: Springer-Verlag, 2009, pp. 111–126.

[4] K. Iwata and T. Kobayashi, "Conversational speech synthesis system with communication situation dependent HMMs," in *Proceedings of the Paralinguistic Information and its Integration in Spoken Dialogue Systems Workshop*, R.-C. Delgado and T. Kobayashi, Eds. New York: Springer, 2011, pp. 113–123.

[5] K. Maekawa, "Production and perception of 'paralinguistic' information," in *Proc. Speech Prosody*, 2004, pp. 367–374.

[6] S. Makino and M. Tsutsui, *A Dictionary of Basic Japanese Grammar*. Tokyo: The Japan Times, Ltd., 1986.

[7] The National Language Research Institute, *Bound Forms ('Zyosi' and 'Zyodôsi') in Modern Japanese: Uses and Examples*. Tokyo: Shuei Shuppan, 1951 [in Japanese].

[8] M. Tsuchihashi, "The speech act continuum: An investigation of Japanese sentence final particles," *J. Pragmatics*, vol. 7, no. 4, pp. 361–387, August 1983.

[9] H. M. Cook, "The sentence-final particle *ne* as a tool for cooperation in Japanese conversation," in *Japanese/Korean Linguistics*, H. Hoji, Ed. Stanford: The Stanford Linguistics Association, 1990, vol. 1, pp. 29–44.

[10] A. Kamio, "The theory of territory of information: The case of Japanese," *J. Pragmatics*, vol. 21, no. 1, pp. 67–100, January 1994.

[11] S. K. Maynard, *Japanese Communication: Language and Thought in Context*. Honolulu: University of Hawai'i Press, 1997.

[12] H. Saigo, *The Japanese Sentence-Final Particles in Talk-in-Interaction*. Amsterdam: John Benjamins Publishing Co., 2011.

[13] Y. Asano-Cavanagh, "An analysis of three Japanese tags: *Ne*, *yone*, and *daroo*," *Pragmatics & Cognition*, vol. 19, no. 3, pp. 448–475, 2011.

[14] N. Yoshizawa, "Intoneeshon (Intonation)," in *A Research for Making Sentence Patterns in Colloquial Japanese 1: On Materials in Conversation*. Tokyo: Shuei Shuppan, 1960 [in Japanese], pp. 249–288.

[15] T. Moriyama, "Bun no imi to intoneeshon (Sentence meaning and intonation)," in *Kooza Nihongo To Nihongo Kyooiku 1: Nihongogaku Yoosetsu*, Y. Miyaji, Ed. Tokyo: Meiji Shoin, 1989 [in Japanese], pp. 172–196.

[16] T. Koyama, "Bunmatsushi to bunmatsu intoneeshon (Sentence-final particles and sentence-final intonation)," in *Speech and Grammar*, Spoken Language Working Group, Ed. Tokyo: Kurosio Publishers, 1997 [in Japanese], pp. 97–119.

[17] S. Kori, "Intoneeshon (Intonation)," in *Asakura Nihongo Kooza 3: Onsei On'in (Asakura Japanese Series 3: Phonetics, Phonology)*, Z. Uwano, Ed. Tokyo: Asakura Publishing Co., Ltd., 2003 [in Japanese], pp. 109–131.

[18] Y. Katagiri, "Dialogue functions of Japanese sentence-final particles *'yo'* and *'ne'*," *J. Pragmatics*, vol. 39, no. 7, pp. 1313–1323, July 2007.

[19] E. Ofuka, J. D. McKeown, M. G. Waterman, and P. J. Roach, "Prosodic cues for rated politeness in Japanese speech," *Speech Communication*, vol. 32, no. 3, pp. 199–217, October 2000.

[20] H. Tanaka, *Turn-Taking in Japanese Conversation: A study in grammar and interaction*. Amsterdam: John Benjamins Publishing Co., 1999.

[21] C. T. Ishi, "The functions of phrase final tones in Japanese: Focus on turn-taking," *J. Phonetic Soc. Japan*, vol. 10, no. 3, pp. 18–28, December 2006.

[22] V. Aubergé, T. Grépillat, and A. Rilliard, "Can we perceive attitudes before the end of sentences? The gating paradigm for prosodic contours," in *Proc. EUROSPEECH*, 1997, pp. 871–874.

[23] V. J. van Heuven, J. Haan, E. Janse, and E. J. van der Torre, "Perceptual identification of sentence type and the time-distibution of prosodic interrogativity markers in Dutch," in *Proc. ESCA Workshop on Intonation*, 1997, pp. 317–320.

[24] M. Sugito, "Shuujoshi 'ne' no imi, kinoo to intoneeshon (Meanings, functions and intonation of sentence-final particle 'ne')," in *Speech and Grammar III*, Spoken Language Working Group, Ed. Tokyo: Kurosio Publishers, 2001 [in Japanese], pp. 3–16.

[25] J. J. Venditti, K. Maeda, and J. P. H. van Santen, "Modeling Japanese boundary pitch movements for speech synthesis," in *Proc. 3rd ESCA/COCOSDA Workshop on Speech Synthesis*, 1998, pp. 317–322.

[26] Y. Greenberg, N. Shibuya, M. Tsuzaki, H. Kato, and Y. Sagisaka, "A trial of communicative prosody generation based on control characteristic of one word utterance observed in real conversational speech," in *Proc. Speech Prosody*. PS8–8–37, 2006.

[27] K. Iwata and T. Kobayashi, "Expressing speaker's intentions through sentence-final intonations for Japanese conversational speech synthesis," in *Proc. Interspeech*. Mon.P2b.03, 2012.

[28] S. Fujie, Y. Matsuyama, H. Taniyama, and T. Kobayashi, "Conversation robot participating in and activating a group communication," in *Proc. Interspeech*, 2009, pp. 264–267.

[29] H. Kawahara, I. Masuda-Katsuse, and A. de Cheveigné, "Restructuring speech representations using a pitch-adaptive time-frequency smoothing and an instantaneous-frequency-based F0 extraction: Possible role of a repetitive structure in sounds," *Speech Communication*, vol. 27, no. 3–4, pp. 187–207, April 1999.

[30] J. H. Ward, Jr., "Hierarchical grouping to optimize an objective function," *J. Am. Statistical Assoc.*, vol. 58, no. 301, pp. 236–244, March 1963.

Unified numerical simulation of the physics of voice. The EUNISON project.

Oriol Guasch[1], *Sten Ternström*[2], *Marc Arnela*[1] *and Francesc Alías*[1]

[1]GTM Grup de recerca en Tecnologies Mèdia,
La Salle, Universitat Ramon Llull, Barcelona, Catalonia, Spain
[2]Department of Speech, Music and Hearing, Kungliga Tekniska Hgskolan, Stockholm, Sweden
oguasch@salleurl.edu, stern@csc.kth.se, marnela@salleurl.edu, falias@salleurl.edu

Abstract

In this demo we will briefly outline the scope of the european EUNISON project, which aims at a unified numerical simulation of the physics of voice by resorting to supercomputer facilities, and present some of its preliminary results obtained to date.

Index Terms: Voice production, finite element methods, physics of voice

1. Demo description

Several strategies have been followed in the past decades to simulate the human voice. These usually focus on the final goal of generating a realistic acoustic signal of voice, making whatever simplifications may be necessary for it. For instance, current commercial speech synthesis systems are based on concatenation of pre-recorded audio segments, whereas many articulatory physics-based approaches rely on one-dimensional formulations that need to incorporate several tricks, so as to emulate three dimensional (3D) voice effects.

However, the amazingly growing capacity of computers combined with intense research on numerical mathematics, has opened the door to go one step beyond. It is at the core of the European FET project EUNISON (Extensive UNIfied-domain SimulatiON of the human voice, http://www.eunison.eu), to build a new voice simulator which, from given inputs representing topology or muscle activations or phonemes, it will render the complex 3D physics of the voice, including of course its acoustic output. This will give important insights into how the voice works, and how it fails.

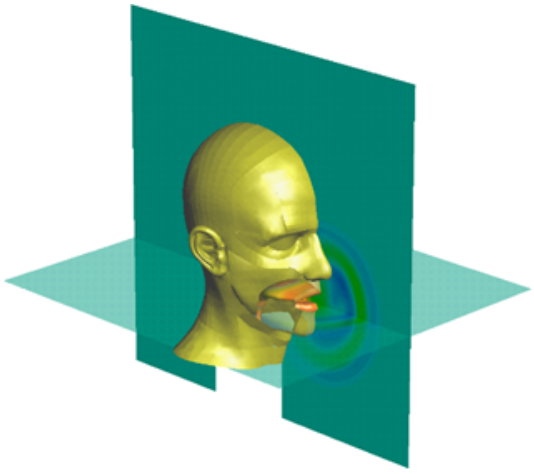


Figure 1: *Finite Element simulation for vowel /a/. Courtesy of La Salle R&D, Universitat Ramon Llull, Barcelona, Catalonia, Spain.*

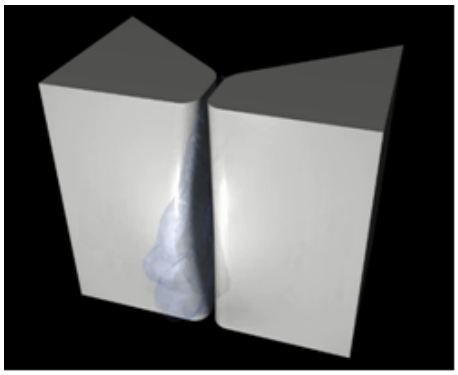


Figure 2: *Vocal folds numerical simulation. Courtesy of the Computational Technology Laboratory, Royal Institute of Technology (KTH), Stockholm, Sweden.*

The physics of voice is extremely intricate. It involves complex interactions between laminar and turbulent airflow; vibrating, deforming, colliding elastic solids; and sound waves resonating in a contorting, dynamic duct. The simulation of these phenomena demands tailored numerical strategies which will be accounted for within EUNISON by adopting and developing new stabilized finite element methods (FEM) to be implemented in parallel platforms. Yet, no good progress can be made without testing the numerical simulations against experimental data. Another crucial aspect of the project will be then to perform

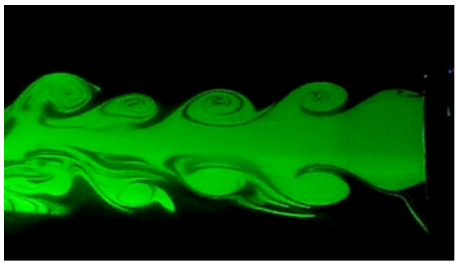


Figure 3: *Experimental flow for a fricative sound. Courtesy of the GIPSA-lab, Centre National de la Recherche Scientifique (CNRS), Grenoble, France.*

experimental investigation, which will not only serve for testing purposes, but also to better understand several mechanisms of voice production related e.g., to the generation of fricatives, plosives and syllables.

In this demo we will present some preliminary results attained at the initial steps of EUNISON (2013-2016). These will consist of several animations and figures corresponding to simulations of some particular phenomena that ideally, by the end of the project, will be integrated in a single framework. In Fig.1 we can observe a snapshot of acoustic waves emanating from a realistic real head when pronouncing vowel /a/. Fig.2 presents a snapshot of the self-oscillation of vocals folds driven by air emanating from lungs, which already includes the possibility of vocal folds contact. In Fig.3, we show the experimental visualization of flow corresponding to a fricative consonant.

The final goal of EUNISON is not developing a text-to-speech synthesis system, but rather a voice simulation engine, with many applications; given the right controls and enough computer time, it could be made to speak in any language, or sing in any style. The model will be operable on-line, as a reference and a platform for others to exploit in further studies. The long-term prospects include more natural speech synthesis, improved clinical procedures, greater public awareness of voice, better voice pedagogy and new forms of cultural expression.

2. Acknowledgements

This research is supported by EU-FET grant EUNISON 308874.

Mage - HMM-based speech synthesis reactively controlled by the articulators

Maria Astrinaki[1], *Alexis Moinet*[1], *Junichi Yamagishi*[2,3], *Korin Richmond* [2], *Zhen-Hua Ling*[4], *Simon King*[2], *Thierry Dutoit*[1]

[1]Circuit Theory and Signal Processing Lab, Numediart Institute, University of Mons, Belgium
[2]The Centre for Speech Technology Research, University of Edinburgh, Edinburgh, UK
[3]National Institute of Informatics, Tokyo, Japan
[4]University of Science and Technology of China (USTC), China

maria.astrinaki@umons.ac.be, alexis.moinet@umons.ac.be, jyamagis@inf.ed.ac.uk
korin@cstr.ed.ac.uk, zhling@ustc.edu, simon.king@ed.ac.uk, thierry.dutoit@umons.ac.be

Abstract

In this paper, we present the recent progress in the MAGE project. MAGE is a library for realtime and interactive (reactive) parametric speech synthesis using hidden Markov models (HMMs). Here, it is broadened in order to support not only the standard acoustic features (spectrum and f_0) to model and synthesize speech but also to combine acoustic and articulatory features, such as tongue, lips and jaw positions. Such an integration enables the user to have a straight forward and meaningful control space to intuitively modify the synthesized phones in real time only by configuring the position of the articulators.
Index Terms: speech synthesis, reactive, articulators

1. Reactive HMM-based speech synthesis

MAGE is based on HTS [1], which it extends in order to support realtime architecture and multithreaded control. MAGE uses multiple threads, and each thread can be affected by the user. This allows accurate and precise control over the different production levels of the artificial speech. MAGE integrates three main threads: the *label thread* responsible for the contextual control, the *parameter generation thread* responsible for the reactive parameter generation by means of short-term parameter trajectories and local maximization and the *audio generation thread* responsible for the vocoding. Three queues are shared between threads for sharing and exchanging data: the *label queue*, the *parameter queue* and the *sample queue*. Further details can be found in [2].

2. Reactive articulatory feature control

In this work, MAGE is modified in order to generate and alter articulatory features. Given the unified acoustic-articulatory model [3], [4] and a set of phonetic labels, it is possible to reactively generate the target speech samples. Simultaneously, it is possible to influence the generated acoustic features by replacing the generated articulatory features with the user input. In this way, we can achieve the goal of altering the generated speech samples at the articulatory level rather than directly at the acoustic level. Note that the intention is to reactively alter a given context and its acoustic features by using only modifications over the articulatory features provided by the user. Here we present an application that combines the MAGE synthesizer with a graphical user interface (GUI)[1]. The GUI is dependent on the database we use for the synthesis [3]. It depicts a two dimensional midsagittal view of the vocal tract drawn using 124 points.The position of these EMA points can be reactively controlled by the user using a mouse or touch screen. There are no limits to the possible position of the EMA points providing to the user 12 degrees of freedom. This means that the user is free to place these points in coordinates that are "unnatural" either from a physical point of view or as sequence of movements. The user is also able to load predefined configurations of vocal tract shapes and EMAs and continue to apply his own controls. The shape of the vocal tract can be reactively altered so that the user will have a reference point to the initial configuration chosen. The final part of the application, generating the speech waveform, is implemented by MAGE. The user modifications over the EMA points are taken into account to generate the corresponding articulatory features. These features are used to estimate the acoustic features, then will give the final speech samples.

3. Conclusions

We presented a method that enables reactive articulatory control over HMM-based parametric speech synthesis using MAGE combined with an application that enables the user to reactively control the position of the articulators through a GUI. We see that reactive articulatory control is feasible, and combined with an interface allows us to explore different aspects of the speech production. However the most important aspect of this work is that we actually prove that MAGE can be used also as a reactive mapping tool between different feature streams. In this case MAGE is able to reactively map the user controls over the articulatory feature stream over the acoustic feature stream.

4. References

[1] H. Zen, K. Tokuda, and A. W. Black, "Statistical parametric speech synthesis," *Speech Communication*, vol. 51, pp. 1039–1064, 2009.

[2] M. Astrinaki, N. d´Alessandro, L. Reboursière, A. Moinet, and T. Dutoit, "MAGE 2.00: New features and its application in the development of a talking guitar," in *Proc. of NIME´13*, 2013.

[3] Z.-H. Ling, K. Richmond, J. Yamagishi, and R.-H. Wang, "Integrating articulatory features into HMM-based parametric speech synthesis," *IEEE TASLP*, vol. 17, no. 6, pp. 1171–1185, 2009.

[4] Z. Ling, K. Richmond, and J. Yamagishi, "Articulatory control of HMM-based parametric speech synthesis using feature-space-switched multiple regression," *IEEE TASLP*, vol. 21, no. 1, pp. 207–219, 2013.

[1]A video demonstration of the presented system can be found in https://vimeo.com/67404386.

Reactive accent interpolation through an interactive map application

Maria Astrinaki[1], *Junichi Yamagishi*[2,3], *Simon King*[2], *Nicolas d´Alessandro*[1], *Thierry Dutoit*[1]

[1]Circuit Theory and Signal Processing Lab, University of Mons, Belgium
[2]The Centre for Speech Technology Research, University of Edinburgh, Edinburgh, UK
[3]National Institute of Informatics, Tokyo, Japan

maria.astrinaki@umons.ac.be, jyamagis@inf.ed.ac.uk, Simon.King@ed.ac.uk
nicolas.dalessandro@umons.ac.be, thierry.dutoit@umons.ac.be

Abstract

MAGE enables the reactive and continuous models modification in the HMM-based speech synthesis framework. Here, we present our first prototype system for extended interpolation applied for interactive accent control. Available accent models for American, Canadian and British English are manipulated in realtime by means of a gesturally controlled interactive geographical map. The accent interpolation is applied to one gender at a time, but the user is able to reactive alter between genders, while controlling the speakers to be interpolated at a time.
Index Terms: speech synthesis, reactive, dialect, interpolation

1. Reactive HMM-based speech synthesis

In the application, various English accents need to be controlled and interpolated in realtime. Therefore, we use MAGE[1], which supports a realtime architecture for reactive HMM-based speech synthesis. MAGE uses multiple threads, and each thread can be affected by the user, allowing accurate controls over the different production levels of the artificial speech [1]. Accessing and controlling the thread responsible for the model manipulation we can reactively modify the way the available models are used for the parameter generation. MAGE allows the reactive control of the the interpolation weights of every feature stream for every phonetic label, as illustrated in Figure 1. This feature allows reactive and continuous control over the degree of interpolation between various models, maintaining any other controls.

2. Reactive accent interpolation map

In order to separate out speaker characteristics and accent so that listeners can focus only on accent transitions we use multiple speakers who have similar accents, by interpolating their acoustic models. As users interact with the map application[2] they selected the speakers for interpolation. All speakers are chosen from the CSTR voice banking corpus.

The application consists of the world map, on which every single speaker is represented as a circle. The "active" region controlled by the user is represented as a yellow circle around the cursor. The user can zoom in/out, navigate, select the speaker's gender and the interpolation "mode" by using the standard mouse or touchscreen controls. There are two ways to interpolate between speakers: "collision" mode, where the active region overlaps and selects one or more speaker for interpolation and "continuous" mode, where each time the cursor moves, the distance between the cursor and all the available

[1]MAGE : http://mage.numediart.org/.
[2]A video can be found in https://vimeo.com/67662099.

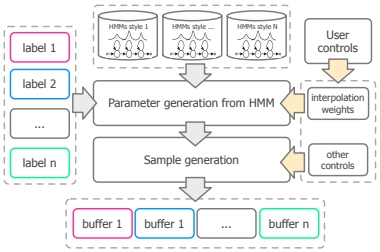


Figure 1: Reactive parameter generation by continuously interpolating multiple models.

speakers is computed and the N-nearest speakers are selected to be interpolated as shown in Figure2. When the voice models are selected, the interpolation weights are computed (uniform weights of $w = 1/N$), the speech parameters are generated and the speech output is synthesized.

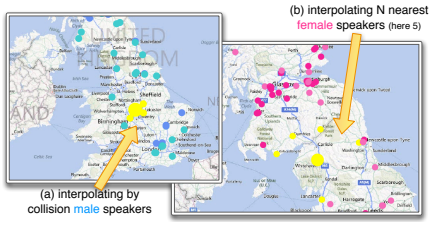


Figure 2: Examples of interactive accent interpolation during (a) "collision" and (b) "continuous" modes.

3. Conclusions

The interactive accent map application can have several applications, targeting the creation of unique personalized voices. In the field of new interfaces for musical expression and performing arts, in gaming, movie dubbing or GPS applications as well as assistive applications for speech impaired people. Finally in speech pedagogy and therapy by creating adaptive references for certain dialects [2]. However it is not straightforward to formally evaluate the proposed interactive accent control.

4. References

[1] M. Astrinaki and N. d´Alessandro and L. Reboursière and Alexis Moinet and T. Dutoit, MAGE 2.00: New Features and its Application in the Development of a Talking Guitar, Proc. of NIME´13.

[2] M. Tachibana, J. Yamagishi, T. Masuko, and T. Kobayashi, Speech Synthesis with Various Emotional Expressions and Speaking Styles by Style Interpolation and Morphing, IEICE Transactions, E88-D, 11, 2484–2491, 2005.

Gesture Control of HMM-Based Singing Voice Synthesis

Christophe Veaux[1], Maria Astrinaki[2], Keiichiro Oura[3], Robert A.J Clark[1], Junichi Yamagishi[1]

[1] Centre for Speech Technology Research (CSTR), University of Edinburgh, UK
[2] TCTS Lab, University of Mons, Belgium
[3] Departement of Computer Science, Nagoya Institute of Technology, Japan
E-mail: {cveaux, jyamagis}@inf.ed.ac.uk

Abstract

The flexibility of statistical parametric speech synthesis has recently led to the development of interactive speech synthesis systems where different aspects of the voice output can be continuously controlled. The demonstration presented in this paper is based on MAGE/pHTS, a real-time synthesis system developed at Mons University. This system enhances the controllability and the reactivity of HTS by enabling the generation of the speech parameters on the fly. This demonstration gives an illustration of the new possibilities offered by this approach in terms of interaction. A kinect sensor is used to follow the gestures and body posture of the user and these physical parameters are mapped to the prosodic parameters of an HMM-based singing voice model. In this way, the user can directly control various aspect of the singing voice such as the vibrato, the fundamental frequency or the duration. An avatar is used to encourage and facilitate the user interaction.

Index Terms: Performative Speech Synthesis, Mage, Singing Voice Synthesis

1. Introduction

In typical text-to-speech systems, the conversion between text and generated voice occurs at the sentence level. This precludes any form of real-time interaction and control of the output voice, narrowing down the set of potential application of these systems. However, HMM-based speech synthesis [1] brings a new level of flexibility in the generation of the speech parameters. It allows for instance to change the characteristics of a voice [2] or to interpolate between voice models [3] and therefore constitutes a promising framework to explore interactive applications of speech synthesis. Recently, the University of Mons has introduced a modified version of HTS [4], called performative HTS or pHTS [5]. With this new system, the HMM models and the speech parameters can be modified on the fly, enabling a real-time control of the voice output. The demonstration prototype presented in this paper gives an illustration of the possibilities offered by this approach. It uses the pHTS engine in order to modify in real-time the prosodic parameters of an HMM-based singing voice model [6]. The prosodic parameters are mapped to the gestures and body posture of the user tracked by a Microsoft Kinect sensor [7]. This kind of gesture control is particularly well suited for real-time interaction where several parameters can be controlled at the same time. It also provides a physical experience of several dimensions of the singing voice prosody. The section 2 of this paper presents a short overview of pHTS and its real-time software environment MAGE. The prototype used for gesture-controlled singing voice synthesis will be detailed in section 3.

2. Mage/pHTS

The pHTS engine relies on a series of modifications of HTS, enabling a much more reactive control of speech output. The main modification is the reduction of the phonetic context used for the generation of the speech parameters trajectories. The speech parameters are generated using a 2-label sliding window and the corresponding speech sound can be synthesized right away as shown in Figure 1. Within appropriate real-time audio software architecture, it means that the sound can be synthesized on the fly. As a result, any kind of modification performed on the models during the generation of the speech parameters has an impact on the corresponding speech sound with a delay of only one label.

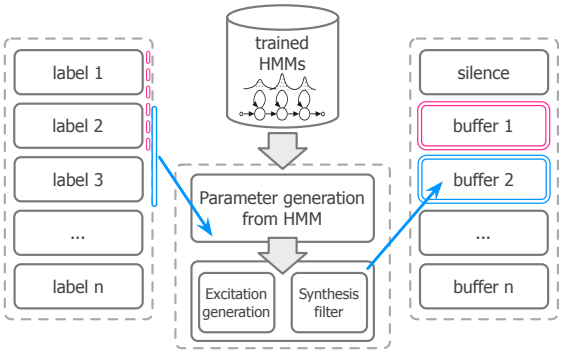


Figure 1: pHTS synthesis using 2-label sliding window to generate the speech parameters.

MAGE is the software umbrella that provides the appropriate real-time audio architecture for the pHTS engine. It also provides a user-friendly API and can be embedded in the PureData programming environment as an external.

3. Gesture control of singing voice synthesis

The MAGE real-time library has already been used in various prototypes exploring how HMM-based speech synthesis can be controlled by gestures. The concept of gesture is here considered in a very large sense, including finger gestures [8] as well as mouth expressions [9]. We present here a new prototype for the gesture control of HMM-based singing voice synthesis. This prototype is based on the following main components, as shown in Figure 2:

- Kinect sensor and its Natural User Interface (NUI) library.
- PureData, real-time programming environment.
- Mage/pHTS, real-time synthesis library.
- Animata, real-time avatar animation software.

The NUI runtime library makes it possible to recognize and track the skeleton joints of a user in front of the Kinect sensor. The skeleton joints coordinates are send to PureData through Open Sound Control (OSC) messages. A series of PureData patches convert the absolute joints coordinates into relative positions and normalizes them by the body size of the user. Finally the gesture descriptors are used to modify the prosodic parameters of the HMM-based singing voice through the MAGE API. The skeleton joints coordinates are also sent to the real-time avatar animation software.

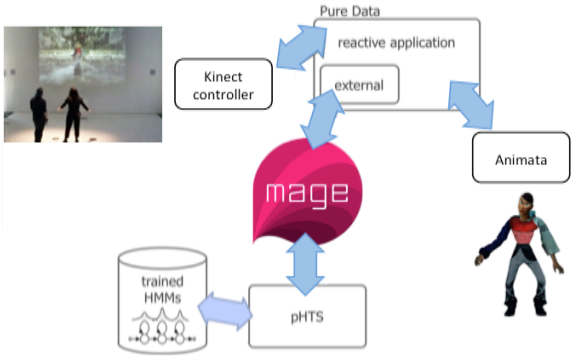


Figure 2: Layout of the gesture-controlled singing synthesis.

The MAGE API was slightly modified in order to enable the control of the following prosodic parameters:

- Vibrato
- Fundamental frequency F0
- Singing speed
- Vocal tract length VTL
- Voicing / Breathiness

The voicing / breathiness parameter was initially introduced for the real-time control of speech synthesis and is not used as a control for the singing synthesis. The mapping between the other prosodic parameters and the hands positions is illustrated in Figure 3.

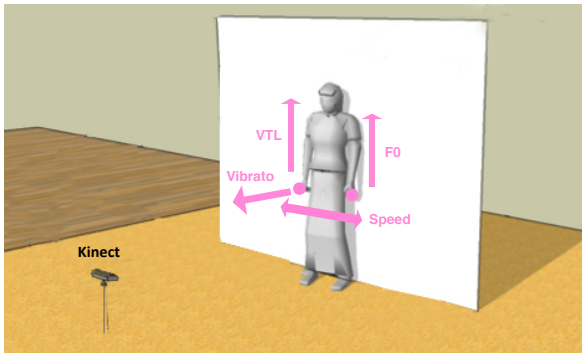


Figure 3: Mapping of the prosodic parameters.

This demonstration prototype has been displayed in various public spaces in Edinburgh as it provides a ludic approach to speech / singing synthesis for the general public.

4. Conclusions

We presented a demonstration prototype based on Mage/pHTS where different prosodic parameters of an HMM-based singing voice can be continuously controlled. This prototype has been designed originally to provide a ludic approach to speech synthesis for the general public but we believe that interactive speech synthesis has a large potential of useful applications. To mention one of them, we envision the use of interactive speech synthesis for voice-output communication aids that could generate spontaneous speech with minimal delay and allow the patient to modify the output speech on the fly.

5. References

[1] T. Yoshimura, K. Tokuda, T. Masuko, T. Koyabashi and T. Kitamura, “Simultaneous modeling of spectrum, pitch and duration in HMM-based speech synthesis”, in IE-ICE Transactions on Information and Systems, vol 83, no 11, pp. 2347-2350, 1999.

[2] T. Masuko, K. Tokuda, T. Koyabashi and S. Imai, “Voice characteristics conversion for HMM-based speech synthesis system”, vol 3, IEE, pp. 1611-1614, 1997.

[3] T. Yoshimura, K. Tokuda, T. Masuko, T. Koyabashi and T. Kitamura, “Speaker interpolation in HMM-based speech synthesis system’, IEE, pp. 2523 -2526, 1997.

[4] H. Zen, K. Tokuda and A.W. Black, ‘Statistical parametric speech synthesis’, Speech Communication, vol 51, no 11, pp. 1039-1064, 2009.

[5] M. Astrinaki, N. d’Alessandro, B. Picart, T. Drugman, T. Dutoit. “Reactive and Continuous Control of HMM-based Speech Synthesis”. in Proc. Speech and Language Technology, Miami, USA, Dec. 2012.

[6] K. Oura, A. Mase, T. Yamada, S. Muto, Y. Nankaku, and K. Tokuda, “Recent Development of the HMM-based Singing Voice Synthesis System – Sinsy”, 7th Workshop on Speech synthesis, 2010, Japan.

[7] Z. Zhang, Microsoft Kinect Sensor and Its Effect, IEEE Multimedia Magazine, vol. 19, no. 2, pp. 4-10, 2012.

[8] “MAGE and HandSketch”, available at: http://vimeo.com/39558917”, March 2012.

[9] “MAGE and Face Tracking”, available at: http://vimeo.com/39567236”, March 2012.

SASSC: A Standard Arabic Single Speaker Corpus

Ibrahim Almosallam, Atheer AlKhalifa, Mansour Alghamdi,
Mohamed Alkanhal, Ashraf Alkhairy

The Computer Research Institute
King Abdulaziz City for Science and Technology
Riyadh, Saudi Arabia

{ialmosallam,aalkhalifa,mghamdi,alkanhal,alkhairy}@kacst.edu.sa

Abstract

This paper describes the process of collecting and recording a large scale Arabic single speaker speech corpus. The collection and recording of the corpus was supervised by professional linguists and was recorded by a professional speaker in a soundproof studio using specialized equipments and stored in high quality formats. The pitch of the speaker (EGG) was also recorded and synchronized with the speech signal. Careful attempts were taken to insure the quality and diversity of the read text to insure maximum presence and combinations of words and phonemes. The corpus consists of 51 thousand words that required 7 hours of recording, and it is freely available for academic and research purposes.

Index Terms: Text-to-Speech, Arabic Speech Corpus

1. Introduction

In the last few years, an increasing number of research projects in Natural Language Processing (NLP) have developed an interest in the Arabic language. Arabic is the official language in more than 21 countries and has two major forms: Standard Arabic and Dialectical Arabic. While dialectical Arabic is the native language of many Arabic speakers nowadays, Standard Arabic is the formal language used in education, culture and media. Standard Arabic consists of: Modern Standard Arabic (MSA) and Classical Arabic. Though MSA is derived from Classical Arabic , the language of Qura'n (Islam's Holy Book), it is more simplified [1].

This paper describes the process of building an Arabic speech corpus, with the aim of collecting large amounts of Arabic speech data for research purposes. The availability of such corpora without copyrights restrictions is important for Arab and non-Arab researchers who are looking for speech resources of contemporary Arabic. MSA was the language of choice to represent the literary standard across the Arab world, while the existing of an Arabic corpora that are specified for speech synthesis and recognition research purposes is still insufficient. Despite the existence of several Arabic speech corpora developed in the last few years, a diverse fully transcribed and segmented MSA speech corpus is a necessity for Text-To-Speech (TTS) and Automatic Speech Recognition (ASR) research and development.

The presented corpus was designed to contain a large selection of text collected from different modern text resources. These resources were chosen to capture the phonetic distribution of the Arabic language. Although the overwhelming majority of the corpus is in MSA form, a designated subcategory was recorded in traditional or classical Arabic. The data was then revised and manually diacritized (Tashkeel) by linguists. Afterward, the voice and pitch of the speaker were captured using specialized equipments and recorded in high quality formats by a professional speaker. The transcriptions were revised again to match the pronunciation of the spoken records. The recordings were then divided into phrases by each sentence's pause and stored in separate files. Finally, the data was phonetically segmented and labeled in preparation for TTS and ASR usages.

The paper is organized as follows: In section 2, we present and discuss some related work. Section 3 outlines the constitution of the corpus and section 4 will describe the recording procedure. Then, the corpus evaluation procedure will be described in section 5 . Finally, in section 6 we conclude and discuss the availability of the dataset.

2. Related Work

The quality of speech synthesis systems highly depends on the corpus's phoneme set distribution. According to [1], an overlook on the existing Arabic speech datasets shows the lack of available public domain, raw-data and single speaker MSA corpus. Similar corpora are either copyrighted, accented [2] [3] and or of a multi-speaker [4]. Most available datasets are designed for speech recognition proposes, whereas TTS systems mostly rely on commercial corpora or record the designated speech database according to the system's needs [5].

The Linguistics Data Consortium (LCD) has a collection of MSA and dialectical Arabic speech and text datasets for research and development purposes. The content varies between annotated news, conversational telephone speech [6] [7] and others; scripted and unscripted while recorded in different conditions. On the other hand, Speech Ocean offers an Arabic speech synthesis database (King-TTS-004) recorded in a studio [8]; It contains a single speaker recordings by a native professional broadcaster with sampling rate of 16kHz and two channels for speech and Electro Glotto Graph (EGG) signals. It includes 12 hours of pure recording time of 3930 sentences with sub database for currency, time, numbers, English alphabets and so on. However, these databases are licensed and can only be used for a fee.

Access to fully available, transcribed and segmented speech corpora is essential in speech and language research. As an example, [9] built 7 speech databases of 7 Indian languages and was released without restriction for commercial and non-commercial usage. The text was selected from Wikipedia articles in Indian languages due to the lack of public domain text corpora. Afterwards, a set of 1000 phonetically balanced sentences were nominated using Festvox script that applies certain

criteria to achieve the optimal selection. The data was recorded by native speakers of these languages in a professional recording studio, and the transcription was altered to accommodate any mistakes were done while recording. The databases were between one to two hours long, and the recordings were automatically segmented afterward using Zero Frequency Filtering (ZFF) technique.

Another important source of data in speech and language research is the Electro Glotto Graph signal or (EGG). EGG , also known as Laryngograph signal, is the measurement of vocal folds vibrations during speech. These measurements are sensed by electrodes attached to the speaker's throat . It is important in aiding medical research such as voice disorders or laryngeal mechanism. Moreover, in NLP it plays an important role in speech recognition and synthesis. For example, it can act as an additional source of information in word recognition [10] or as a mean of detecting glottal pulses (pitch-marks) to assure that the synthesized speech is in a consistent matter.

Likewise, [11] introduced a high-quality, Romanian speech corpus called RSS. It is also available for academic uses to help promote Romanian speech technology research. The data was recorded at 96 kHz sampling frequency then down-sampled to 48 kHz based on an over-sampling method with the intention of noise reduction. However, the total length of the recorded data is only about 3.5 hours based on 3500 sentences. Additionally, the article discussed the effect of sampling frequency on the speaker similarity of synthesized voices. It found that the use of lower sampling frequencies such as 16kHz lowers the similarity of the synthesized voice to the original speaker.

3. Constitution of the Corpus

3.1. Text Selection

The text corpus included a variety of genres and writing styles from multiple sources to insure a representative sample. Furthermore, diversity of pronunciation was also taken into consideration during the text selection process to capture the entire phoneme spectrum that exist in the Arabic language. The corpus is divided into sub-categories to capture a wide variety of vocabulary and to enable users of the data set to perhaps extract "exact" relevant phrases if necessary. For example, the use of numbers, date and time are very common in most applications. The dataset is divided into nine such categories:

1. Dates and Time: Phrases about time such as dates, time of the day, days of the week and months of the year, In both Hijri and Gregorian calendars .
2. Numbers: The speaker uttered several numbers in isolations as well as in combined forms. For example the number 123 was pronounced as (one two three) and as (one hundred twenty three).
3. Financial: Phrases about money and currency.
4. Customer Service: Phrases common in IVR systems (question and answer format).
5. Names: A list of common Arabic names (first, middle and last).
6. Story: Excerpts from novels and stories,
7. Traditional: Text selected from classic sources such as old books.
8. Miscellaneous: A collection of phrases from various domains and writing styles
9. News: Excerpts from local and foreign news

	Total	Unique	Median
Syllables	333,981	627	2
Tri-phones	324,225	8,689	7
Bi-phones	335,769	1,076	38
Mono-phones	347,313	38	5,651.5

Table 3: Syllables, Tri-phones, Bi-phones and Mono-phones frequencies

The text corpus consists of 51,432 words, amongst which 21,556 were unique and required 7 hours and 20 minutes of audio recording. It contained 627 unique syllables out of a total of 333,981 syllables. The break down by category is provided in table 1. All of the above mentioned categories were recorded in a normal tone of voice. However, a part of the miscellaneous text was recorded using different expressions such as sadness, joy, surprised and questioned for 15 minute each.

3.2. Transcription

A professional linguist was closely involved during both corpus selection and recording to verify the quality of the text and pronunciation. Furthermore, unlike most other languages, Arabic can only be unambiguously pronounced when diacritized. Due to this problem, most applications rely on automatic Arabic diacritization as a pre-processing step to transcription [12]. Therefore, the text was manually diacritized and revised to insure unambiguity. An automatic text-to-transcription algorithm was used to transcribe the text [13]. However, the entire corpus was revised manually. The total number of phonemes used to transcribe the corpus is 38, shown in table 2. The Statistics about the phonemes' and syllables' distribution in the corpus is provided in table 3. Furthermore, the frequency distribution for the phonemes are provided in figure 1.

4. Recording Procedure

4.1. Speaker selection

Due to the nature and size of the corpus, a professional speaker who can maintain his performance for long recording sessions was hired to record the corpus. Although diacritization reduces ambiguity, it makes the reading experience more challenging. Diacritized text forces the reader to pay closer attention to the symbols at the character level which slows down the reading process and makes it more difficult. In normal situations, readers would predict the words they are reading based on the content and experience. Ten candidates were screened and tested to insure their proficiency in the Arabic language, clarity of pronunciation and voice. The panel of linguists voted on the first two criteria while the pleasantness of the voice was aided by online voting. The speaker, selected based on the previous criteria, is a professional male news anchor who has worked in several TV programs and has experience in recording poetry books.

4.2. Recording environment

The recording sessions took place in a sound proof studio to minimize undesired noise. Both the speech and the EGG signals were recorded and stored synchronized in a stereo wave file (left=speech, right=EGG) using a 96 kHz sampling rate and a 16-bit resolution. The intensity of the signal was carefully monitored to remain equal across different sessions by request-

	Number of Words	Unique Words	Recording Time	Number of Utterances
Dates & Time	900	227	00:09:06	226
Numbers	967	133	00:13:17	224
Financial	2080	272	00:21:22	451
Customer Service	2643	928	00:22:58	400
Names	5503	1451	00:43:05	887
Story	4085	2490	00:44:42	151
Traditional	7797	4027	01:10:06	485
Miscellaneous	10927	7025	01:45:48	1011
News	16530	8208	02:44:16	537
Full Corpus	51432	21556	07:20:28	4372

Table 1: General statistics about the text and audio corpora in each sub-category

Symbol	Phone	Symbol	Phone	Symbol	Phone	Symbol	Phone
a	(فتحة) ـَ	gh	غ	m	م	T	ط
A	(فتحة مفخمة) ـَ	h	هـ	n	ن	th	ث
aa	ا	i	(كسرة) ـِ	pau	pause	TH	ظ
Aa	(مد مفخم) ا	I	ي	q	ق	u	(ضمة) ـُ
Ah	ح	j	ج	r	ر	U	و
Az	ذ	Jn	ء	R	ر	w	و
b	ب	JU	ع	s	س	y	ي
d	د	k	ك	S	ص	z	ز
D	ض	kx	خ	sh	ش		
f	ف	l	ل	t	ت		

Table 2: Phonemes and their corresponding labels in the transcription files

ing the speaker to repeat a phrase and compare it with a reference phrase recorded at the first session. The positions of the microphone, the speaker and the EGG electrodes were adjusted in every session to match the reference phrase. A linguist was present to follow the speaker and correct him in case he skipped, mispronounced or could not recognize a word. Each session was 15 minutes long and no more than four sessions per day were taken.

4.3. Editing and preparation

The recorded data was further edited to remove unwanted segments such as errors in pronunciation, long pauses, etc. The sessions were then split based on pauses and the EGG signals were exported into separate files. The corpus is divided into utterances by pauses and an HMM aligner was built to label and segment each utterance on the phoneme level. The Cambridge University HMM toolkit was used to build the speaker-specific aligner [14]. The phoneme symbols, the beginning and the ending timestamps are provided in the transcription files. Although the corpus was segmented and aligned automatically by an HMM system, over half an hour of speech was manually aligned by computational linguists. The final corpus is divided into four sets of files:

1. Wave: The speech waveform for each utterance in 96 kHz and 16-bit resolution.
2. EGG: The corresponding EGG signal in 96 kHz and 16-bit resolution.
3. Text: The corresponding diacritized text.
4. Label: The corresponding transcription along with the phoneme boundary segmentation.

The above set of files are provided for each expressive version and each utterance was labeled by their corresponding category discussed in section 3. The labels are provided in two formats (mono and full) in accordance with the HTS standard format [1]. The mono labels provide only the mono-phoneme sequence along with the beginning and ending timestamps. The full labels however provide the penta-phone sequence, each phone with its two proceeding and two preceding phonemes. Moreover, they are more detailed and provide more information such as the number of phonemes, syllables and words in each utterance. The position and number of phonemes, syllables and words are also indicated in the full label file. In the case of syllables, the files also indicate whether a syllable is stressed or accented. It is worth mentioning here that there are two types of syllables in the Arabic language: open syllables CV and CV:, and closed syllables CVC, CV:C and CVCC. Where C stands for consonant, V for vowel and V: for long vowel.

5. Corpus Evaluation

In order to evaluate the quality of the final dataset, the corpus was used to build a text-to-speech model. This model will put to test various quality aspects of the data, such as: size, structure, format and consistency. This section will describe briefly the TTS model generation and output evaluation procedures. It is important to emphasize that the purpose of the test and evaluation is not to compete with other Arabic TTS systems but to evaluate the dataset and to illustrate its usage.

The HMM-based Speech Synthesis System (HTS) was used to build the model. HTS is a free tool to build HMM-based TTS, which is basically a patch code for the HTK. It provides scripts and code that uses HTK as well as other free speech tools to train TTS models. The version used to evaluate the dataset in this paper was HTS 2.2. The format of the dataset was de-

[1] HMM-based Speech Synthesis System (HTS): http://hts.sp.nitech.ac.jp/

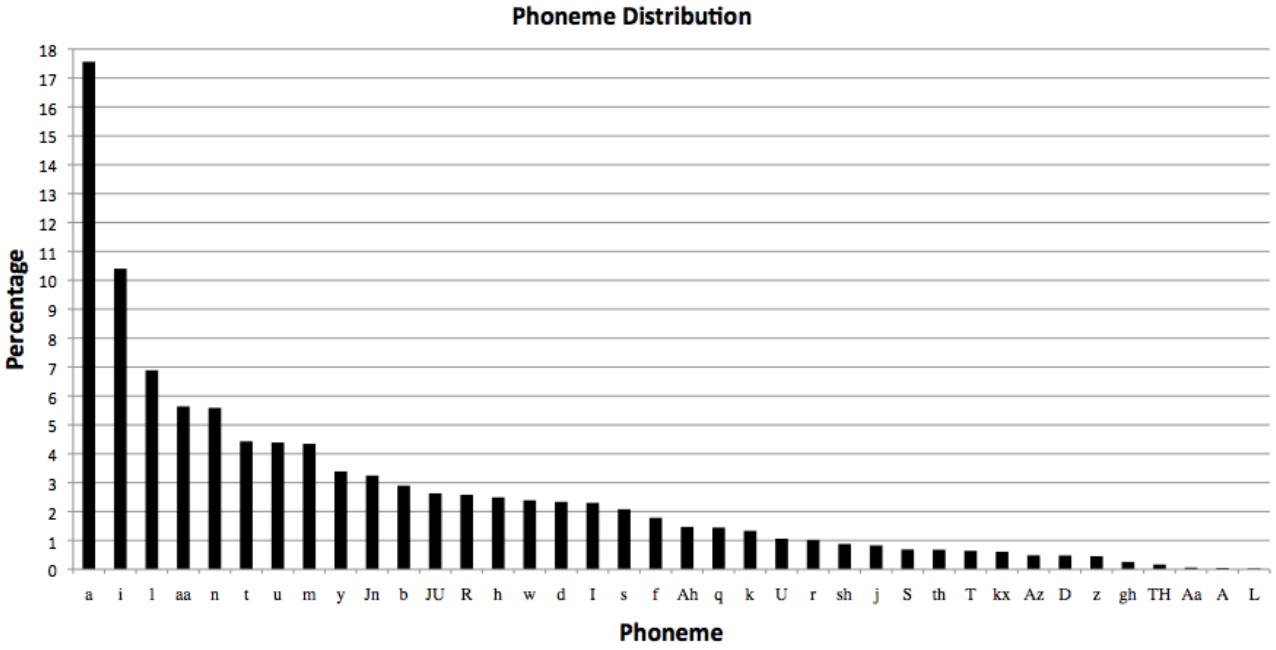


Figure 1: Distribution of the Arabic phonemes in the corpus

signed to be compatible with HTS and the default parameters were tuned for the data. The training procedure completed successfully and five sentences were synthesised to be evaluated. Two well known metrics were measured to evaluate the synthesized speech, namely naturalness and intelligibility. Some of the group members as well as other researchers (total of 10) were asked to listen to the five generated samples and provide their evaluation on each one. Evaluators were asked individually to write down what they have heard, and to rate each audio clip (on a scale from 1 to 5) based on the following qualities:

1. Naturalness: How close was the synthesised speech to being natural? (1: Very robotic sound - 5: Close to natural)
2. Intelligibility: How much effort was taken to understand the content of the sample? (1: Had to focus - 5: Easy to understand)

The average naturalness score was 3.58 and the average intelligibility score was 3.9. The original sentences were compared to what the evaluators have written and the word-error rate was 2.13%. The numbers suggest that the performance of the TTS system on the provided corpus was good on naturalness and very good on intelligibility. The synthesized samples used in the evaluation process can be found at http://cri.kacst.edu.sa/SASSC/samples.zip.

6. Conclusion and Future Work

In this paper, the processes for collecting and recording the SASSC dataset was described. The text was designed to capture a wide variety of usages and as much of the Arabic phonetic spectrum as possible. The text is fully diacratized, transcribed and was recorded in a high quality format in a soundproof studio by a professional speaker along with the pitch signal. Moreover, a TTS system based on the corpus has been evaluated which achieved promising results.

SASSC as large repository for Arabic speech, provides several avenues for future work. These may include a study on the minimum amount of speech required to produce an acceptable voice. Invest more in recording different speech expressions and evaluate their results. Moreover, linguists can find the corpus as a resource for studying different Arabic structures and pronunciations of MSA.

In order to promote and facilitate research and development for Arabic language in the NLP field, the corpus will be freely available for research and educational purposes only, a permission has to be obtained from KACST for any commercial or non-academic use of the corpus. The dataset is available at http://cri.kacst.edu.sa/SASSC/index.html

7. Acknowledgment

The authors would like to acknowledge and thank Dr. Mohsen Rashwan, Dr. Sherif Abdou, Dr. Ahmad Ragheb and Mostafa Shahin from Research and Development International (RDI) at Cairo, Egypt for providing their help and expertise in various phases during the corpus collection and recording procedures.

8. References

[1] N. Habash, *Introduction to Arabic Natural Language Processing*. Morgan and Claypool Publishers, 2010.

[2] M. Alghamdi, F. Alhargan, M. Alkanhal, A. Alkhairy, M. Eldesouki, and A. Alenazi, "Saudi accented arabic voice bank," *Experimental Linguistics ExLing*, p. 9, 2008.

[3] G. Droua-Hamdani, S. A. Selouani, and M. Boudraa, "Algerian arabic speech database (ALGASD): Corpus design and automatic speech recognition application," *Arabian Journal for Science and Engineering*, vol. 35, no. 2C, p. 158, 2010.

[4] M. Algamdi, "KACST Arabic phonetics database," in *The 15th International Congress of Phonetics Science*, 2003, pp. 3109–3112.

[5] F. Chouireb and M. Guerti, "Towards a high quality arabic speech synthesis system based on neural networks and residual excited vocal tract model," *Signal, Image and Video Processing*, vol. 2, no. 1, pp. 73–87, 2008.

[6] M. Maamouri, D. Graff, and C. Cieri, "Arabic broadcast news transcripts," LDC - Linguistic Data Consortium, December 2006. [Online]. Available: http://www.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDC2006T20

[7] Appen-Ltd, "Levantine Arabic conversational telephone speech, transcripts," LDC - Linguistic Data Consortium., Jan 2007. [Online]. Available: http://www.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDC2007T01

[8] SpeechOcean, "Arabic speech synthesis database I (Male)," April 2013. [Online]. Available: http://www.speechocean.com/en-TTS-Corpora/371.html

[9] K. Prahallad, N. Kumar, V. Keri, R. S, and A. W. Black, "The IIIT-H indic speech databases," in *INTERSPEECH*, 2012.

[10] P. S. Dikshit and R. W. Schubert, "Electroglottograph as an additional source of information in isolated word recognition," in *Proceedings of the Fourteenth Southern Biomedical Engineering Conference*, 1995, pp. 1–4.

[11] A. Stan, J. Yamagishi, S. King, and M. Aylett, "The Romanian speech synthesis (RSS) corpus: building a high quality HMM-based speech synthesis system using a high sampling rate," *Speech Communication*, vol. 53, no. 3, pp. 442–450, 2011.

[12] M. Rashwan, M. Al-Badrashiny, M. Attia, S. Abdou, and A. Rafea, "A stochastic Arabic diacritizer based on a hybrid of factorized and unfactorized textual features," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 1, pp. 166–175, 2011.

[13] M. Attia, "Theory and implementation of a large-scale arabic phonetic transcriptor, and applications," Ph.D. dissertation, Dept. of Electronics and Electrical Communications, Cairo University, Cairo, Egypt, 2005.

[14] S. Young, G. Evermann, M. Gales, D. Kershaw, G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev, and P. Woodland, "The htk book version 3.4," *Cambridge University Engineering Department*, 2006.

Prosodically Modifying Speech for Unit Selection Speech Synthesis Databases

Ladan Golipour, Alistair Conkie, Ann Syrdal

AT&T Shannon Laboratory, Florham Park, NJ, USA

{ladan,adc,syrdal}@research.att.com

Abstract

This paper investigates the practical limits of artificially increasing the prosodic richness of a unit selection database by transforming the prosodic realization of constituent sentences. The resulting high-quality transformed sentences are added to the database as new material.

We examine in detail one of the most challenging prosodic transformations, namely converting statements into yes/no questions. Such transformations can require very large prosodic modifications while at the same time there is a need to retain as much naturalness of the signal as possible.

Our data-driven approach relies on learning templates of pitch contours for different stress patterns of interrogative sentences from training data and later on applying these template pitch contours on unseen statements to generate the corresponding questions.

We examine experimentally how the modified signals contribute to the perceived synthesis quality of the resulting database when compared with baseline unmodified databases.

Index Terms: speech synthesis, RELP, prosody

1. Introduction

Unit selection synthesis [6] can generate very natural audio output but output quality may not be consistent, depending largely on the voice and size and audio quality of the database used. Best quality output is generally produced when synthesizing in-domain text.

In this paper we examine one method of addressing some of the limitations of unit selection synthesis by means of prosodically enriching the underlying speech database.

Descriptions of several techniques for enhancing the quality of unit selection databases have been published previously, each with its own merits. They tend to focus mostly on the segmental level. The possibility of substituting units generated by a formant synthesizer was examined in [4], [5].

There have also been efforts concerned with using data from other voices to boost the effective size of a database [2], [1]. By combining voices, the overall coverage is improved along with the resulting synthesis quality.

One limitation of unit selection synthesis is that generally no prosody modification is performed. A practical consequence is that at times a desired prosodic contour for a synthetic sentence will be unavailable and substituted by a less satisfactory sequence of units. The above mentioned techniques do not tackle this issue directly.

Some interpolation or averaging techniques have been developed, for example [9], [10] where a fusion approach is employed if specific appropriate units are not available.

Our approach in this paper is to create new prosodic patterns and to add them as extra data to the database. The extra data are based on utterances already in the database, but have been transformed using signal processing to have a different prosodic realization, e.g. as a question rather than as a declarative sentence. In particular, we systematically examine taking selected sentence or phrase elements from the unit selection database and modifying them prosodically, converting nuclear pitch accents from H* to L* and end tones from L-L% to H-H% in ToBI [11] formalism, then augmenting the database with the modified data.

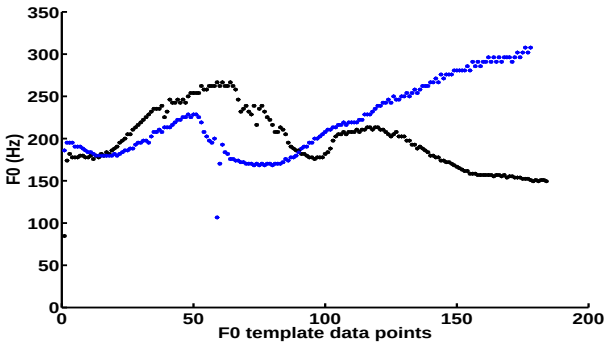


Figure 1: F0 values for a statement/question pair with stress pattern 10010.

We expect to increase the prosodic coverage of the database while maintaining quality. As part of this process we describe a robust data-driven technique for speaker-specific prosody prediction.

As an example, the prosody of the declarative sentence "Calling Michael Jordan." would typically have an high (H*) pitch accent on the first syllable of Jordan and a low end tone (L-L%), The question form as in "Was that Michael Jordan?" would typically have an low (L*) pitch accent on the first syllable of Jordan and a high (H-H%) end tone.

One challenge is the signal modification required for the task. As can be seen from Figure 1, the degree of modification needed is large (on the order of an octave phrase finally) and potentially can introduce artifacts into the signal, reducing naturalness. For this work, we used and tested Residual Excited Linear prediction (RELP) [7] and Pitch Synchronous Overlap and Add (PSOLA) [8], in combination with prosodic templates learned from data.

Additionally, there are challenges relating to data labeling accuracy, and to scaling up techniques to work effectively with large datasets that are not manually labeled.

2. Methodology and Dataset

We first describe how new prosody was generated from existing utterances using a data-driven approach. To achieve this we used a specially constructed dataset. The speech database was recorded from a female speaker of American English under controlled conditions, and is part of a larger set of recordings

designed for various synthesis experiments. The audio files are 16kHz, 16 bit audio. The prosody dataset is composed of approximately 2100 sentence pairs of the form "Calling Robert Kerr." and "Was that Robert Kerr?". Each pair uses a different combination of first name and last name. One from each pair has a declarative intonation and one a yes-no interrogative.

We randomly split the data into a training set with 1600 example pairs and a test set with 500 example pairs. We extracted only the first and last name portion of the examples using their transcriptions. All the examples are categorized based on their syllable stress pattern. Syllables with primary stress are marked *1* and all other syllables are marked *0*. Ignoring word boundaries, this leads to 10 stress-pattern classes for the data, with the distribution among classes shown in Figure 2. For example, the most common pattern, 1010 could represents names like *Richard Johnson* or *Jane Andreesen*. Next, we trained target

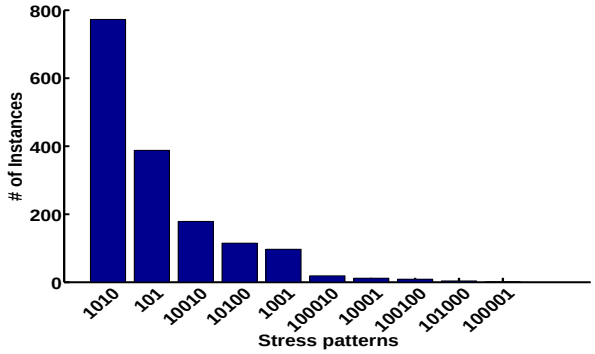


Figure 2: Frequency count of the 10 stress pattern classes in the training data.

prosody templates as described in detail below. The templates were used to generate hypotheses that can then be compared with the reference data both objectively and subjectively.

We examine multiple methods to convert the declarative pronunciation form of names to their interrogative form. In all approaches, we attempt to achieve this goal initially through pitch modification. In the first approach, we employ the PSOLA algorithm[8]. In the second approach, we decompose the speech signal into residual and LPC coefficients using one implementation of the RELP algorithm [7]. Finally, we use RELP-PSOLA [3], where PSOLA operates on the residual signal, in order to reduce the amount of distortion that PSOLA introduces to the speech signal, especially when there is a large difference between the pitch in the original and the target speech signal.

2.1. Pitch Template Computation

The first step in all approaches was to estimate a template pitch contour for the interrogative form. We observed that the shape of the pitch contour largely depends on the stress pattern of the name. Different names with similar stress patterns have similar pitch contours. Taking account of this, we categorized the interrogative training examples according to their stress pattern and estimated an average pitch contour for each category. Figure 3 displays pitch contour templates for the 10 stress patterns. In order to generate the pitch contour templates, we performed the following procedure for every stress category:

- Generate pitch marks for all interrogative training examples using the RELP algorithm and form a pitch vector from the pitch duration values.
- Rank all pitch vectors based on their length and choose the median pitch vector as the class's reference.
- Apply dynamic time warping (DTW) on all pitch vectors in order to align them with the reference pitch vector.
- Compute the mean of the aligned pitch vectors. The mean vector is still not a smooth representation of a pitch template contour due to occasional errors in the pitch marks and the performance of the DTW algorithm.
- Perform one-dimensional median filtering on the mean pitch vector to generate a smoother pitch contour template.

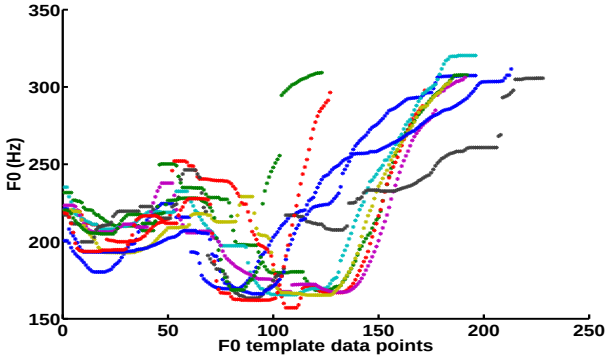


Figure 3: Interrogative pitch contour templates for different stress patterns

In this way, an interrogative pitch contour for every stress-pattern is generated. Next, we employ this contour to modify the pitch values of declarative sentences according to their stress patterns.

2.2. Declarative to Interrogative Conversion based on PSOLA

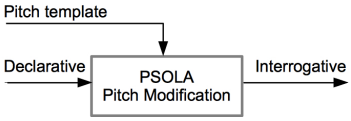


Figure 4: PSOLA conversion.

The block diagram of the first pitch modification approach based on PSOLA is depicted in Figure 4. The interrogative pitch contour is represented as a vector of pitch values. In order to use it with the PSOLA algorithm, we resampled this vector through interpolation in such a way that the summation of all pitch values in the final pitch vector is approximately equal to the length of the test example. Next, we aligned the first pitch mark of the template with the first pitch mark of the test example, and used PSOLA to modify the pitch of the example. Sometimes there is a large change in pitch value, particularly for the final syllables, for which PSOLA is not an ideal choice. Motivated by this we also examined RELP-based approaches.

2.3. Declarative to Interrogative Conversion based on RELP

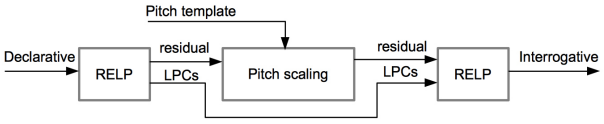


Figure 5: RELP conversion.

In the second approach, we decomposed the speech signal into residual and LPC coefficients and perform a simple modification of pitch marks through resampling the residual. The block diagram of this method is shown in Figure 5. We applied RELP to perform the decomposition, and then extracted the pitch marks from the residual signal. We resampled the pitch template vector in order to have the same number of pitch marks as the residual signal, and computed the ratios between template pitch values and test example pitch values for every adjacent pitch mark. Finally, we resampled the residual signal using this vector of ratio factors and reconstructed the hypothesis speech signal with the modified residual signal and original LPC coefficients. Since the higher pitch values (towards the end of question form) affect the length of the speech output and hence the local speech rate, we compensate for this phenomenon through resampling the speech and keeping the duration of the output speech approximately equal to the original signal.

2.4. Declarative to Interrogative Conversion based on RELP and PSOLA

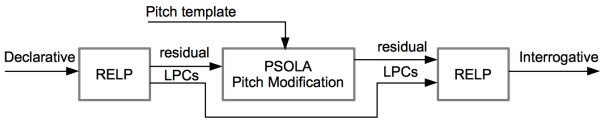


Figure 6: RELP and PSOLA

In the third approach, we combined the PSOLA- and RELP-based techniques. Instead of applying PSOLA to the test example using a template pitch contour, we decomposed the speech signal into residual and LPC envelope using RELP and applied the pitch contour on the residual signal using PSOLA. Next, we reconstructed the signal with RELP using the modified residual and the original LPC coefficients. This approach is similar to the previous one with the advantage that here, the pitch modification of the residual signal is achieved through a more sophisticated algorithm rather than a simple resampling technique. Also, the number of pitch marks does not necessarily need to be equal in the original and modified residual. If PSOLA repeats a frame, we simply use the same LPC coefficients and if it drops the frame, we ignore the LPC coefficients for that frame. Another advantage of this approach is that since PSOLA is used only to modify the residual signal, the amount of distortion introduced to the target signal is potentially less than the PSOLA-only approach.

The algorithms described above are relatively sensitive to the segmental time-alignment accuracy of the database. The phoneme boundaries for the source and target data are a result of running forced alignment recognition on the speech data and are relatively (but certainly not completely) accurate. Boundaries that are inaccurate have the potential to affect the quality of the imposed pitch curves. In practice, while this was a concern and was monitored, it did not seem to strongly impact synthesis quality for the examples we chose. However it could be a concern where large portions of a database are being processed.

Figure 7 displays pitch values for two reference examples and their relative pitch-modified hypotheses generated by the second RELP-based approach. As can be seen, the predicted pitch contours are very similar to their reference counterparts. Voiced and unvoiced data are treated differently in terms of pitch marks, so accurate matching of voiced to voiced frames and unvoiced to unvoiced frames in source and target were carefully monitored.

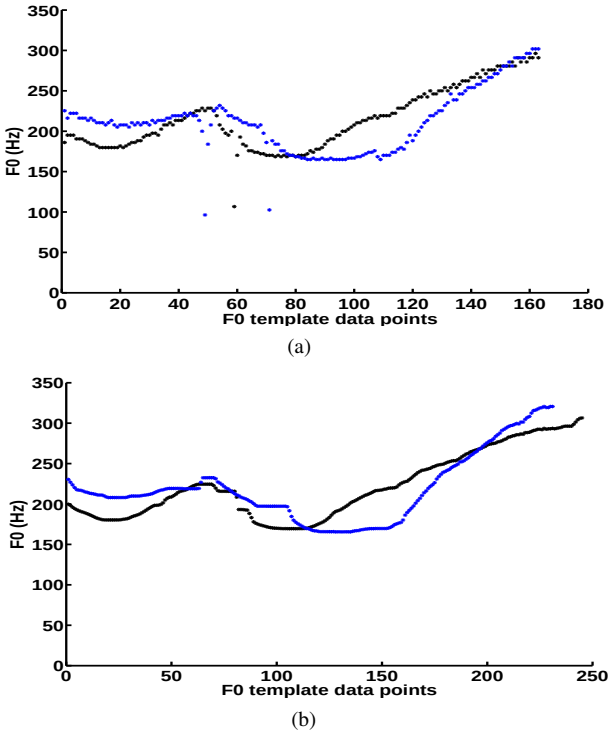


Figure 7: F0 values of (a) reference natural speech data and (b) RELP-based pitch-modified data for two sentences.

3. Experiments

The main goal of this work is to extend the database of a speech synthesizer through adding prosodically-modified units (based on existing units) to the database. These extra units can be selected and concatenated like any others in the database, potentially leading to a more natural pitch contour in the synthesized speech. We designed an experiment to test this idea. Before we carried out the main experiment described below, we performed an informal evaluation of the three approaches (PSOLA, and the two RELP variations) by listening to approximately 100 examples from each approach. We concluded that RELP-based approaches have similar performance while the the PSOLA-based approach has a slightly worse performance. We decided to include only the second RELP-based approach in the subjective experiment.

The next step was to prepare voices for the synthesis experiment. Here we describe the three synthesized database compositions. The three databases have in common 45 minutes of recordings designed to be diphone-rich. The voices differ with respect to the type of name-specific data they contain. The *high baseline* contains 55 minutes of natural recordings of complete interrogative sentences (carrier phrases and names). The *low baseline* has no interrogative sentences. The *RELP-based* voice contains 40 minutes of the RELP-based data: declarative names-only sentences (no carrier phrase) that have been converted to interrogative form using signal processing.

Once the voices were built, our standard unit selection synthesizer was used to generate stimuli for testing. We chose a set of 20 example sentences of the form "Was that James Baxter?" and synthesized them with the three different voices, giving a total of 60 stimuli. It is important to note that in choosing the sentences we used a best-case approach and preferred samples where the signal-processed audio was generated with the correct prosody by the unit selection. Figure 8 demonstrates the unit

selection patterns of test examples generated using the RELP-based voice. As expected, the units for the carrier phrase ("Was

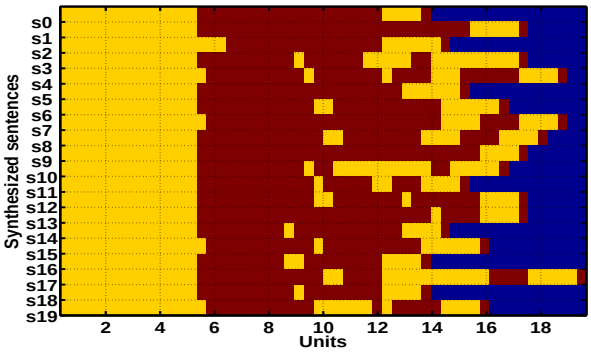


Figure 8: Units selected by the synthesizer for 20 test examples are colored based on their origin. Yellow represents "general data" and red "RELP-based pitch-modified data". Blue represents the silence that follows the end of each sentence.

that") are all extracted from the general data while a large portion of units for the name pronunciations originate in the RELP-based data. However, the synthesizer seems to select many general data units towards the end of the sentence except for a few final units for which the pitch is in its highest values. This is mainly due to the synthesis algorithm and the tradeoffs between respecting prosody requests and achieving appropriate transitions at unit boundaries.

The 60 stimuli were presented to listeners in the form of a web based test. The stimuli were presented in a different random order for each listener. Listeners were asked to judge each of the audio samples on a scale of 1 to 5 where the following descriptive terms were used 1 (Bad); 2 (Poor); 3 (Fair); 4 (Good); 5 (Excellent). There were a total of 17 listeners. Listeners were only asked to rate the sentences and were not given any specific instructions about naturalness or about focusing on prosody. There were two supplementary questions to determine (a) whether or not the listener was a native speaker of English, and (b) whether the listener heard the audio via headphones or via loudspeaker.

4. Results and Discussion

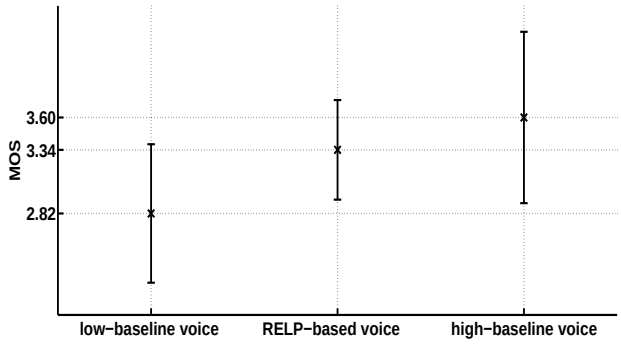
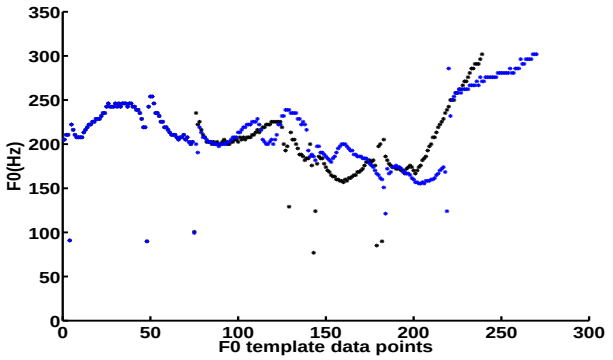
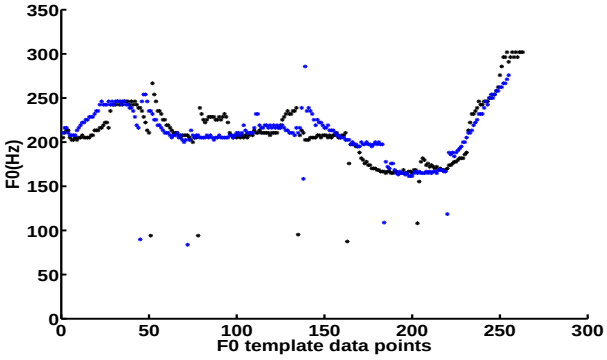


Figure 9: MOS with standard deviations for the three voices used in the web-based test.

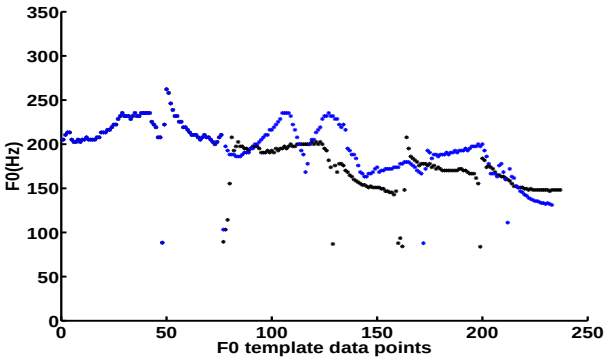
The results of the web-based test are depicted in Figure 9. The highest Mean Opinion Score (MOS) for the test, 3.60, is achieved by the high baseline system, while the low baseline score is 2.82. The RELP-based interrogative names database score of 3.34 is intermediate between the baseline conditions. The differences between all conditions are statistically significant ($p < 0.005$).



(a)



(b)



(c)

Figure 10: F0 values for two examples of synthesized speech. (a) High baseline voice. (b) RELP-based voice. (c) Low baseline voice.

Pitch contours of two test examples synthesized with three voices: high baseline, low baseline, and RELP-based are shown in Figure 10. Both high baseline and RELP-based voices successfully show a rising pitch contour towards the end of the sentence, while the low baseline voice lacks the yes-no question-form pitch contour. This is expected as no interrogative units exist in the database of the low baseline voice.

The results indicate that there was a definite overall benefit from including the prosodically-modified data in the database, but that it is not, as configured currently, a perfect substitute for including unmodified recordings. It is however encouraging that the value for the signal-modified database is closer to the high baseline than to the low baseline.

5. Conclusions and Future Work

In this paper we have shown that it is possible to use signal processing techniques to modify speech signals considerably, even far outside the typical +/- 20% range that is recommended in the literature, in order to create prosodically distinct variants of sentences. We achieved our goal by means of prosodic templates derived from a training set. We describe using these techniques to create questions from statements.

The prosodically modified data was added to our unit selection database and used at synthesis time. Effectively we have increased the size, and most importantly, the prosodic coverage of the database. The techniques are tested here on name data since we have natural target speech available, but can be applied in principle to any form of prosody modification where some parallel prosodic data is available for training.

The experimental results indicate that enriching a database prosodically, even with material that has been subjected to signal processing manipulations, can benefit synthesis quality significantly.

Future work will extend the ways in which we can generate new prosody contours for material to be added to the unit selection database.

6. References

[1] A. Conkie, and A. Syrdal, "Composite TTS Voices", 7th Speech Synthesis Workshop 2012, Kyoto, Japan.

[2] E. M. Eide, and M. A. Picheny, "Towards pooled-speaker concatenative text-to-speech", Proc. of ICASSP 2006.

[3] B. Gold, N. Morgan, D. Ellis, "Speech and Audio Signal Processing: Processing and Perception of Speech and Music".

[4] S. R. Hertz, I. C. Spencer, and R. Goldhor, "When can speech segments serve as surrogates?", Presentation at From Sound to Sense: 50+ Years of discoveries in Speech Communication 2004.

[5] S. R. Hertz, I. C. Spencer, and R. Goldhor, "Perceptual consequences of nasal surrogates in English: Implications for speech synthesis", 147th meeting of the Acoustical Society of America, 2004.

[6] A. Hunt, and A. Black, "Unit selection in a concatenative speech synthesis system using a large speech database", Proc. of ICASSP 1996, pp. 373-376.

[7] D. T. Magill, and C. K. Un, "Speech Residual Encoding by Adaptive Delta Modulation with Hybrid Companding", Proc. of National Electronics Conference 1974, pp. 403-408.

[8] E. Moulines, and F. Charpentier, "Pitch-synchronous waveform processing techniques for text-to-speech synthesis using diphones", Speech Communication 9, pp. 453-467.

[9] T. Okubo, R. Mochizuki, T. Kobayashi, 2006. "Hybrid voice conversion of unit selection and generation using prosody dependent HMM", IEICE Trans. Inf. Syst. E89-D (11), 27752782.

[10] M. Tamura, N. Braunschweiler, T. Kagoshima and M. Akamine, "Unit Selection Speech Synthesis Using Multiple Speech Units at Non-adjacent Segments for Prosody and Waveform Generation", IEEE Proc. ICASSP2010, March 2010.

[11] K.E.A. Silverman, M.E. Beckman, J.F. Pitrelli, M. Ostendorf, C.W. Wightman, P. Price, J.B. Pierrehumbert, and J. Hirschberg, "TOBI: a standard for labeling English prosody", in Proc. ICSLP, 1992.

Combining a Vector Space Representation of Linguistic Context with a Deep Neural Network for Text-To-Speech Synthesis

Heng Lu, Simon King, Oliver Watts

The Centre for Speech Technology Research, The University of Edinburgh, UK

hlu2@inf.ed.ac.uk, Simon.King@ed.ac.uk, owatts@staffmail.ed.ac.uk

Abstract

Conventional statistical parametric speech synthesis relies on decision trees to cluster together similar contexts, resulting in tied-parameter context-dependent hidden Markov models (HMMs). However, decision tree clustering has a major weakness: it use hard division and subdivides the model space based on one feature at a time, fragmenting the data and failing to exploit interactions between linguistic context features. These linguistic features themselves are also problematic, being noisy and of varied relevance to the acoustics.

We propose to combine our previous work on vector-space representations of linguistic context, which have the added advantage of working directly from textual input, and Deep Neural Networks (DNNs), which can directly accept such continuous representations as input. The outputs of the network are probability distributions over speech features. Maximum Likelihood Parameter Generation is then used to create parameter trajectories, which in turn drive a vocoder to generate the waveform.

Various configurations of the system are compared, using both conventional and vector space context representations and with the DNN making speech parameter predictions at two different temporal resolutions: frames, or states. Both objective and subjective results are presented.

Index Terms: TTS, speech synthesis, deep neural network, vector space model, unsupervised learning

1. Introduction

Traditionally, text-to-speech (TTS) conversion systems use a carefully-constructed *linguistic specification* as the interface between the text and the waveform. These representations might be created in a number of ways, from the hand-coded rules of a formant synthesizer [1, 2] to the complex multi-layered structures typically found in systems such as Festival [3]. The latter are built up in a number of steps, using numerous models (and a few rules). Each of these models is typically trained in a supervised fashion from labelled data (e.g, speech with hand-labelled phrase breaks). This makes such systems expensive to port to a new language and hard to adapt to a specific application domain.

From the linguistic specification, a waveform is generated. This might be done through the concatenation of short segments of speech, as in the unit selection method, or through a trainable statistical parametric model. The HMM-based statistical parametric speech synthesis method [4, **?**, 5, 6] is the most widely used statistical parametric approach, not least because it has advantages over unit selection such as smaller footprint, stable output, and more flexibility (e.g., speaker adaptation). However, the way in which the linguistic specification is mapped by decision trees onto a set of disjoint context-dependent models may not make the most effective use of training data.

1.1. Representing and modelling linguistic context

In speech synthesis, context-dependent models are necessary to capture contextual effects in speech, including co-articulation and supra-segmental variation. Wide context modelling is much more important than in automatic speech recognition. Current approaches effectively naively multiply out all the context features (e.g., next phone, preceding phone, position in syllable, ... etc) to create a vast state space with a cardinality equal to the product of the cardinalities of all the context features.

1.2. Data fragmentation and averaging

In order learn such a model from data in which only a tiny fraction of possible models actually have training examples, decision tree-based clustering is employed. Controlling the complexity of the resulting clustered model is non-trivial. The standard approach not only effectively fragments the data into disjoint subsets, for estimating the parameters of each context-dependent model[1], yet at the same time averages not just over multiple speech samples but over multiple clustered contexts. This averaging must inevitably lose some of the detail, which may be necessary for natural-sounding speech. This loss of fine detail is thought to be a main contributing factor to the 'muffled' sound of the generated speech.

1.3. The linguistic specification

Another problem is that, although the statistical parametric method learns from speech data using a well-defined objective function, it still relies on the linguistic specification, which is the basis for the context-dependent modelling. A properly-designed question set – for example, using categories of place and manner of articulation – is required when using these context features to cluster the acoustic models. Obtaining the knowledge and data required to construct the linguistic specification, and to properly use it for model clustering, is difficult for under-resourced languages.

1.4. An alternative approach to the linguistic specification: a Vector Space-based Front End

From the field of NLP, we have drawn inspiration from work in which letter and word representations are constructed in a way that is optimal for a certain specific task. [7] describes a self-organizing codebook that is jointly optimized with the text-

[1]Of course, Expectation-Maximisation training actually uses 'soft counting' and not strictly hard assignment of speech frames to single model parameters, but nevertheless each speech frame contributes to the estimate of only a very few model parameters.

to-speech model ensuring that the coding is optimal in terms of overall performance. Experiments showed that performance is improved compared to baseline system using orthogonal letter codes. [8, 9] take the idea of task-based estimation of word embeddings from neural net language modelling and apply it to an array of NLP tasks in a multitask learning framework. From these ideas, we have developed in previous work a Vector Space Model-based approach to constructing the linguistic specification.

[15] describes in detail our approach to the unsupervised construction of representations for context modelling in TTS – here we just give a brief summary of how this approach works. The vector space model (VSM) is well established in Information Retrieval (IR) and Natural Language Processing (NLP) as a way of representing objects such as documents and words as vectors of descriptors. To build vector space models, co-occurrence statistics are gathered in matrix form to produce high-dimensional representations of the distributional behaviour of word and letter types in the corpus. Lower dimensional representations are obtained by approximately factorizing the matrix of raw co-occurrence counts by the application of slim singular value decomposition (SVD). The main advantage of the VSM model over the traditional decision tree is that, instead of querying conventional features of linguistic objects, such as the phonetic class to which a phoneme belongs or the part of speech to which a word belongs, the objective distance between the VSM output values directly represents the similarity of two units. The VSMs are learned in an unsupervised fashion from text: no labeled speech is required.

In that previous work, we employed this novel front end in conjunction with conventional decision tree-based model clustering of HMMs-of-context-dependent-units [15]. However, the recursive hard partitioning of the vector space by the decision tree is not entirely satisfying, since it may lose some of the representational power of the VSM and in particular is unable to take best advantage of any factorial structure in the space. Therefore, we now consider a different way to map from the continuously-valued linguistic specification constructed by the VSM, to the parameters of Gaussians from which we can ultimately generate synthetic speech.

1.5. Deep Neural Networks

Compared with the decision tree clustering method, neural networks may be able to better model the interactions between linguistic and acoustic features, including correlations and factorial behaviour. Neural networks were used in [13] to map between a sequence of phonemes and an acoustic description from which a speech waveform can be generated. More recently, in [14] a DNN is employed to establish a mapping between phoneme-level labels represented as a large set of binary input features (derived from the question set that would otherwise have been used in decision tree clustering), numerical input features (for position and phone/syllable number) and target acoustic features. The DNN predicts output on a frame-by-frame basis.

There are several design choices to be made in creating a DNN-based synthesiser, which are not explored in [14]. So, in this paper we offer some comparisons of different input representations and different timescales for predicting the acoustic features.

We compare the DNN approach to TTS with three kinds of input representation: letter-based binary input, phoneme-based binary input, and a letter-based VSM. The binary input is similar to that in [14], with one binary feature for every possible two-way partition of the linguistic feature space. The original motivation behind the VSM front-end, as alluded to earlier, is an attempt to construct a linguistic specification from text using only unsupervised learning, motivated by a need to build TTS systems for under-resourced languages. The resulting representation comprises a continuous vector space which reflects the distributional properties of the training material.

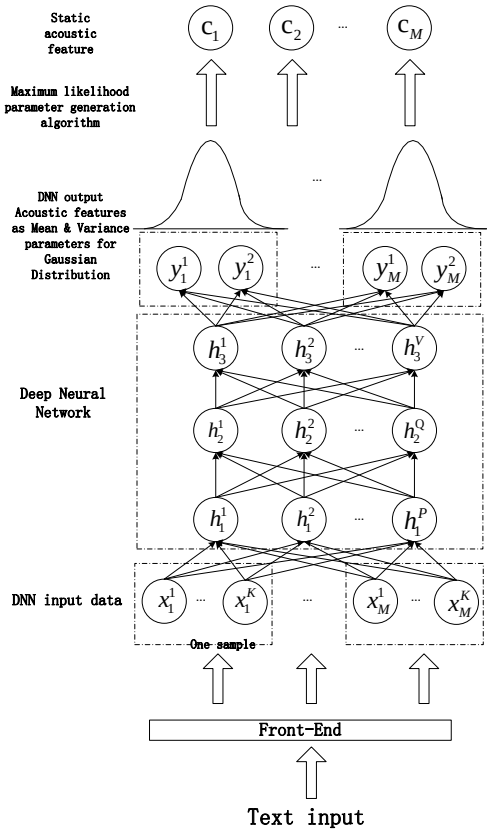


Figure 1: Framework for Deep Neural Networks based TTS

2. System architecture

2.1. Framework

Figure 1 illustrates the framework of the DNN-TTS method. In the training stage, text is first transformed into DNN input labels x_i^j. Where $i = 1, 2, ..., M$ denotes the ith DNN input label vector, and $j = 1, 2, ..., K$ denotes the jth element in the ith DNN input label vector. In this work, both frame-based and state-based systems were trained. In the case of the frame-based DNN, M is set to the total number of frames in the utterance, and the DNN is trained to map from binary labels to acoustic features y_k^l, where k indexes the DNN output vector and l indexes the elements of the output vector. In addition to static spectral envelope features, the acoustic feature vectors also include dynamic features derived from them.

In the case of the state-based DNN, the DNN is trained to map from context-dependent labels to the HMM model parameters for each context-dependent HMM state. The full-context space in speech synthesis is the product of cardinalities of each context feature, and is thus of enormous size.

To summarise, three various representations for the input x_i^j are compared:

1. Phone-based binary features, in which the linguistic context features are converted to a binary representation according to the answers to the conventional set of questions used to build parameter-tying decision trees
2. Letter-based binary features, constructed in an analogous way
3. VSM-based continuous features, as described in section 1.4.

At the synthesis stage, distributions over acoustic features y_k^l are predicted by the DNN from the input features x_i^j for each frame, or for each HMM state. Finally, static acoustic features $\boldsymbol{C} = \{\boldsymbol{c}_1, \boldsymbol{c}_2, ..., \boldsymbol{c}_M\}$ are generated from those distributions using MLPG, and a vocoder is employed to synthesise a waveform from the generated acoustic features. In our experiments, only the means of Gaussian distributions are predicted, and we use pre-computed fixed variances.

2.2. Deep Neural Network Training

In contrast to the data fragmentation inherent in decision tree clustering, a DNN is trained as a single model using all data, via back-propagation, meaning that every training sample effectively contributes to the training of every model parameter (the weights in the DNN). In addition to varying the input representation and temporal granularity, our experiments also varied the number of hidden layers from 1 to 3 layers; we use the weights trained in shallower DNN to initialize the weights for Deeper DNN in the training process. The up to 3 hidden layers have 1000, 500, and 250 units, respectively. And linear layer is used as the output layer.

2.3. Maximum likelihood parameter generation

After generating the acoustic features y_k^l (including both static and dynamic features) from the DNN, we use the maximum likelihood parameter generation (MLPG) method [4] to generate smooth parameter trajectories to drive a vocoder.

3. Experiments

Six systems were built for comparison, as defined in Table 1. Systems C and E are HMM-based benchmark systems that use decision trees to map from the linguistic specification to model parameters; the remainder are DNN systems. Because there are typically more letters than phonemes in a given word or phrase, for the letter-based systems 3 state HMMs are used, whereas for phones 5 state HMMs are used.

3.1. Database

A database of a British English male speaker sampled at 16kHz was used for the experiments. There were 1000 utterances in the database, from which 860 were used for training, and 140 were held out for subjective and objective evaluation. 21-order Line Spectral Pair (LSP) plus delta and delta-delta were extracted to represent the spectrum. Log F0 plus delta and delta-delta were used to model pitch. 10-order features were extracted to model aperiodicity (AP). 3 binary values are used to represent voicing and its delta and delta delta.

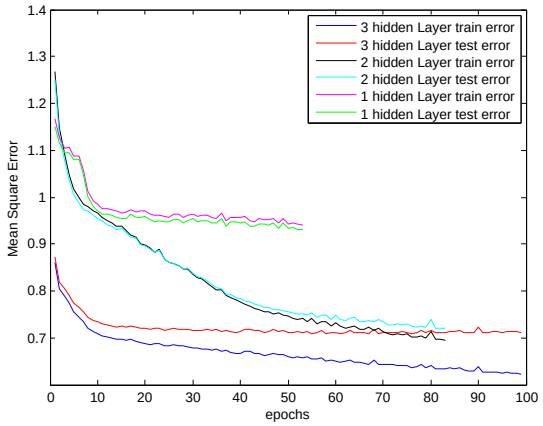


Figure 2: Mean square error for system A (letter-based binary DNN-TTS)

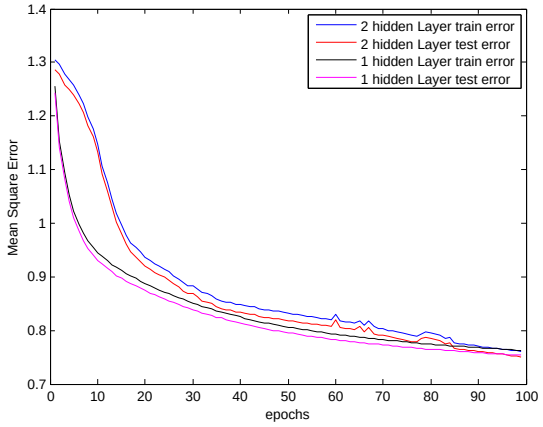


Figure 3: Mean square error for system B (letter-based VSM DNN-TTS)

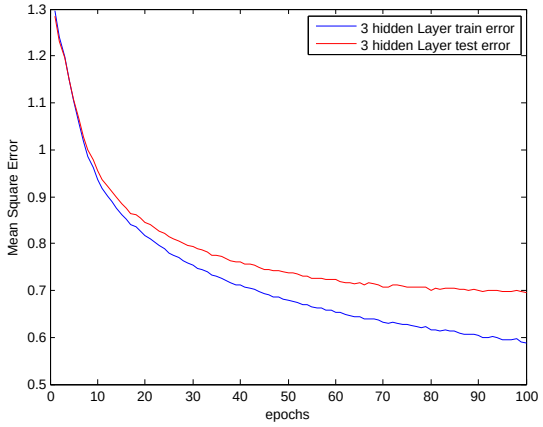


Figure 4: Mean square error for system F (phone-based context-dependent state mapping DNN-TTS)

System	Type	Linguistic unit	Input	Timescale
A	DNN	letter	binary	state
B	DNN	letter	continuous	state
C	Decision tree	letter	labels	state
D	DNN	phone	binary	frame
E	Decision tree	phone	labels	state
F	DNN	phone	binary	state

Table 1: Summary of systems built

3.2. Binary Front end

For the phone-based binary representation, each context-dependent phonetic label is rewritten as a 1256-dimensional binary vector. Features incorporated into this vector include the identities and phonetic categories of the {previous, current, following} phones, as well as word and phrase level positional features. Utterance-level features were excluded in these experiments in order to reduce the input vector dimensionality. In the case of the frame-based DNN, frame position within the current phone (both forwards and backwards) and total length in frames of the current phone are appended as numerical features to the DNN input vector. In the case of the state-based DNN, 5 binary values are appended to the input vector to encode state index. For the binary letter-based representation, letter-related context features are encoded in a 1134-dimensional binary vector for input to the DNN. For the VSM method, each letter in the input text is represented as a 27-dimensional vector of continuous values.

3.3. Objective evaluation

Figures 2 to 4 show the the training and testing error for systems A, B and F respectively. In 2, in can be seen that as the number of DNN hidden layers in increased, error decreases. The plots for the DNNs with 3 hidden layers exhibit some evidence of over-training. In 3, the 2 hidden layer error is not obviously smaller than the 1 hidden layer neural network. This is probably because the input dimension for VSM is only 27, so the DNN does not need so many parameters for the mapping. The training and test error for the 3 hidden layer result is shown in figure 4 for system F, where training and test error is smaller than the equivalent letter-based system A in figure 2. Figure 5 shows MLPG-generated LSP trajectories from system F, compared to natural LSPs. The predicted LSPs lack detail, compared with the natural ones.

Root Mean Square Error (RMSE) between generated LSPs and natural LSPs for a total of 140 utterances were calculated for each system. The results are shown in Table 3. From this result we can see that the two HMM + decision tree baseline systems C and E (letter and phone based respectively) still perform better than the DNN-TTS systems. System F is the best amongst the DNN-based systems, and informal listening confirmed that the voice generated by this system sounds reasonable.

3.4. Subjective evaluation

Six pairwise AB naturalness tests were conducted. 40 utterances were synthesized for each AB test, and 19 native speakers of English were asked to choose the more natural one from each pair. Results are shown in Table 2. All preferences in this Table are significant. The subjective listening test results are in general agreement with the objective measure.

System	A	B	C	D	E	F
A		$>$				$<$
B	$<$		$<$			
C		$>$			$<$	
D						$<$
E			$>$			$>$
f	$>$			$>$	$<$	

Table 2: Subjective listening test results. Blank cells indicate comparisons that were not tested. $>$ indicates that the system named in the row was judged to be better than the system named in the column, and $<$ indicates the reverse.

System	LSP Error
A	0.222398
B	0.228053
C	0.157672
D	0.244260
E	0.135460
F	0.176447

Table 3: Root Mean Square Error for LSP

4. Conclusions

We have presented an attempt to use DNNs to replace decision tree parameter clustering, motivated by a desire to take better advantage of the continuously-valued features produced by our novel VSM-based front-end. From the results obtained, the HMM benchmark systems still outperform the DNNs. It is not surprising that phone-based systems are better than letter-based ones – the point of the letter-based system is not to be better, but rather to be easier to construct for many different languages. The VSM front end does not do as well as hoped, but the comparison with the other systems is not quite fair because the VSM system uses no suprasegmental contextual information. Future work includes refining the input representation further, combining the binary and VSM features into a single system, training the DNN on substantially more data.

5. Acknowledgements

This research was supported by an EPSRC programme grant grant EP/I031022/1 (Natural Speech Technology) and has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement no. 287678 (Simple4All).

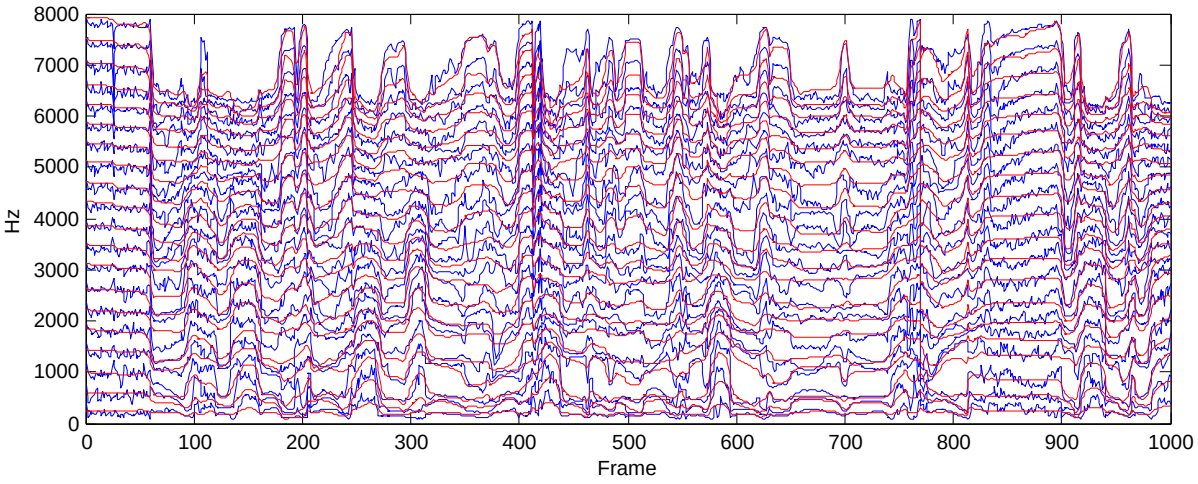


Figure 5: MLPG generated LSP by system F (phones, state units, DNN) in red compared with natural LSPs in blue.

6. References

[1] D. Klatt, "Review of text-to-speech conversion for english," *J. Acoust. Soc. Amer.*, vol. 82, pp. 737–793, 1987.

[2] J. Allen, S. Hunnicutt, and D. Klatt, "From text to speech: The mitalk system," *Cambridge Univ. Press*, 1987.

[3] R. A. J. Clark, K. Richmond, and S. King, "Multisyn: Open-domain unit selection for the Festival speech synthesis system," *Speech Communication*, vol. 49, no. 4, pp. 317–330, 2007.

[4] K. Tokuda, T. Yoshimura, T., Masuko, T. Kobayashi, and T. Kitamura, "Speech parameter generation algorithms for hmm-based speech synthesis," *Proc. of ICASSP*, pp. 1315–1318, 2000.

[5] T. Yoshimura, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, "Simultaneous modeling of spectrum, pitch and duration in hmm-based speech synthesis," *Proc. of Eurospeech*, pp. 2347–2350, 1999.

[6] "HTS," http://hts.sp.nitech.ac.jp/.

[7] Kåre Jean Jensen and Søren Riis, "Self-organizing letter code-book for text-to-phoneme neural network model," in *INTERSPEECH*, 2000, pp. 318–321.

[8] Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel P. Kuksa, "Natural language processing (almost) from scratch," *CoRR*, vol. abs/1103.0398, 2011.

[9] R. Collobert and J. Weston, "A unified architecture for natural language processing: Deep neural networks with multitask learning," in *International Conference on Machine Learning, ICML*, 2008.

[10] G. E. Hinton, S. Osindero, and Y. W. Teh, "A fast learning algorithm for deep belief nets," *Neural computation*, vol. 18, pp. 1527–1554, 2006.

[11] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, pp. 504–507, 2006.

[12] G. Dahl, E. George, Y. Dong, D. Li, and A. Acero, "Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, pp. 30–42, 2012.

[13] O. Karaali, G. Corrigan, and I. Gerson, "Speech synthesis with neural networks," *Proceedings of the 1996 World Congress on Neural Networks*, 1996.

[14] H. Zen, A. Senior, and M. Schuster, "Statistical parametric speech synthesis using deep neural networks," *Proc. of ICASSP*, 2013.

[15] O. S. Watts, "Unsupervised learning for text-to-speech synthesis," *Ph.D. dissertation, University of Edinburgh*, 2012.

Is Unit Selection Aware of Audible Artifacts?

Jindřich Matoušek, Daniel Tihelka, Milan Legát

University of West Bohemia, Faculty of Applied Sciences,
New Technologies for the Information Society,
Univerzitní 8, 306 14, Plzeň, Czech Republic
{jmatouse,dtihelka,legatm}@kky.zcu.cz

Abstract

This paper presents a new analytic method that can be used for analyzing perceptual relevance of unit selection costs and/or their sub-components as well as for tuning of unit selection weights. The proposed method is leveraged to investigate the behavior of a unit selection based system. The outcome is applied in a simple experiment with the aim to improve speech output quality of the system by setting limits on the costs and their sub-components during the search for optimal sequences of units. The experiments reveal that a large number (36.17%) of artifacts annotated by listeners are not reflected by the values of the costs and their sub-componets as currently implemented and tuned in the evaluated system.

Index Terms: speech synthesis, unit selection, concatenation cost, target cost, audible artifacts

1. Introduction

Despite the increasing popularity of HMM based and hybrid speech synthesis methods, unit selection concatenative systems still represent the mainstream in many real life applications, especially in limited domains where synthesized chunks are combined with prerecorded prompts. In such applications, the ability of the unit selection to deliver highly natural and to the recordings well fitting output are the key factors. Not surprisingly, the unit selection also remains the first choice for eBook reading applications, which have been acquiring a lot of interest over the recent years. This is due to a better acceptance of the unit selection synthetic speech output quality by end users.

Nevertheless, the unit selection method has seemed to be getting abandoned as a research topic over the last few years. There is no question that a huge amount of efforts have already been invested in improving its speech output quality since the introduction of the method [1]. It has been analyzed from almost all possible angles. Many works have dealt with experiments introducing different speech parameterizations and distances, which could be used for measuring the quality of concatenations [2], [3]; the target cost sub-components; pruning of the large unit databases; tuning weights of the costs [4]; and last but not least optimizing the unit search to lower computational costs of the method [5], [6], to name some.

Still, we believe that the most important problem related to the unit selection—the haphazard presence of audible artifacts—has not been investigated thoroughly enough. Generally speaking, there are three main sources of these quality drops. First, any database, no matter how thoroughly it is verified, contains mislabelings at different levels. Second, the costs that are used when searching for the optimal sequences of units are not always well correlated with human perception. Third, the traditional implementation of the search algorithm allows, as long as the cost of the whole sequence of units is minimum, for selecting units that should locally be avoided according to their assigned costs. This can especially be observed when the unit database is small. In theory, the same behavior can however be observed in large footprint systems as well.

Little has also been invested in analyzing the audible artifacts and real understanding of the latent constructs that influence human perception of them. This is predominantly a consequence of not having reliable objective methods for TTS quality evaluation as well as large costs and labor intensiveness of the subjective methods. In this paper, we present a method, the goal of which is to provide more insight into the two latter sources of audible artifacts mentioned above. The proposed method represents in its nature an analytic complement to the traditional TTS quality evaluation techniques (e.g. MOS or ABX tests).

The rest of this paper is organized as follows. The next section briefly describes our implementation of the unit selection. We put stress on explaining the individual costs used in our system as they are important for understanding the results of the presented experiments. The proposed analytic method as such does not however depend on their implementation. Section 3 deals with the first perceptual experiment, the goal of which was to detect synthetic units/chunks of utterances that contain audible artifacts. In Section 4, we describe the second perceptual experiment showing to what extend can setting of limits on values of costs and their sub-components im-

prove the quality of the system's output. In Section 5, we briefly discuss the obtained results, and finally, in Section 6, we draw conclusions and outline the intension for our future work.

2. ARTIC TTS System

2.1. Overview

ARTIC (Artificial Talker in Czech) is a Czech text-to-speech system developed since 1997. It is a corpus based system, which makes use of a large carefully designed speech inventory annotated at orthographic, phonetic and prosodic levels. Two speech synthesis methods—fixed-inventory synthesis (a.k.a. diphone synthesis) and unit-selection synthesis using diphones as basic units—had originally been implemented [7]. The system has recently been extended by the HMM-based synthesis [8].

The experiments described in this paper are mainly related to our unit selection implementation. The current target and concatenation cost design is described in the following subsections. The total cost is then a simple sum of the two costs.

2.2. Concatenation Cost Implementation

The concatenation cost consists of three sub-components—the difference in energy, the difference in $F0$ and the Euclidean distance of 12 MFCC coefficients [9]. All the values are z-score normalized in order to align their ranges. Moreover, the $F0$ sub-component is only computed when concatenating diphones at voiced ends. In case that voiced/unvoiced segments are to be concatenated, the $F0$ sub-cost is set to 1. Unvoiced segments are concatenated at zero $F0$ cost. Values of all features are calculated pitch-synchronously and the total concatenation cost is calculated as an average of the values of the three sub-components.

2.3. Target Cost Implementation

To compute the target cost, the following features are evaluated:

- *suitability for prosodic word position*. The feature evaluates the difference in position within prosodic word by a non-linearly increasing penalization [10]. This allows to avoid discrete *initial, middle, final* features and to non-linearly model the positions in a continuous space.
- *type of prosodeme (a sort of a prosodic phrase)* [11]. This feature uses simple binary match criterion.
- *left and right phonetic context*. This feature, also often used as a sub-component of the concatenation cost, penalizes disagreements in left and right phonetic contexts of a given diphone. Similarly to the prosodemes, this feature is binary with all the disadvantages of it. However, some analyses have recently been undertaken to overcome this limitation [12].

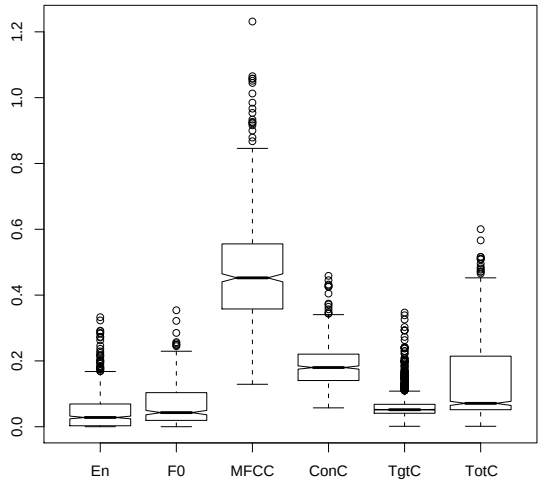


Figure 1: *Boxplots of costs of the units forming the optimal sequences found by the unit selection search algorithm.*

Each feature is weighted by a heuristically set weight (prosodeme the most prominent, the phonetic context the least), and the value of the target cost is then given by the weighted average:

$$TgtC = \frac{\sum_{t=1}^{T} F(t) \times w(t)}{\sum_{t=1}^{T} w(t)}, \qquad (1)$$

where $F(t)$ is the feature value, and T is the number of features.

3. Perceptual Annotation Experiment

3.1. Outlier Detection

As already mentioned in the introduction, this work is aiming at the audible artifacts haphazardly appearing in the output of the unit selection systems. If we start with an assumption that the costs correlate reasonably well with human perception, most of the selected units of extreme costs should lead to audible artifacts.

In order to see whether or not such units are being selected at all, the box-and-whisker diagrams (boxplots) were used. The boxplots of values of all concatenation cost sub-components and also of the costs as such of the units forming the optimal sequences of units in our test set of utterances are shown in Fig. 1. The plots indeed show that some units of rather outlying costs tend to appear in the selected sequences of units.

The goal of the next step is to investigate whether these, in terms of the costs, outlying units coincide with audible artifacts. An "annotation" perceptual experiment

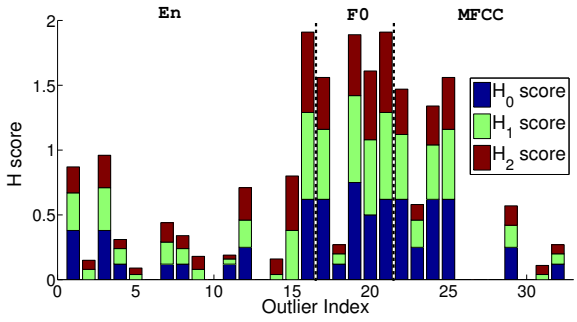


Figure 2: *The H_L scores of the outliers of the concatenation cost sub-components.*

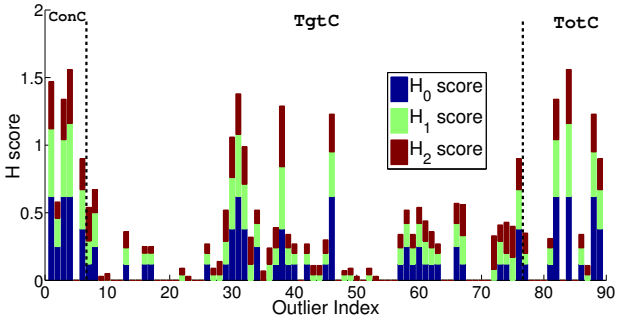


Figure 3: *The H_L scores of the outliers of the unit selection costs.*

was conducted using a set of 50 sentences. At least one unit of an outlying cost or sub-cost was found in approximately 80% of the selected sentences. The test sentences were selected randomly from a large news text corpus. All sentences were manually checked to make sure that they do not contain foreign language inclusions, out-of-vocabulary words or complex tokens that could be wrongly handled by the front end (text analysis) module of our system.

The task of listeners was to mark segments, which they found unnatural or containing any sort of distortion. The shortest segment which could be marked was a phoneme. Most of the participants were typically marking segments of approximate length of 3–5 phonemes. The test was conducted using a web interface allowing the listeners to work from home. It was, however, stressed in the test instructions that the annotations shall be done in a silent environment and using headphones. The listeners were only presented with audio, they did not have access to any visual information like spectrograms or oscilograms of the test sentences. Since the annotation of audible artifacts is not a simple task, only experienced listeners were invited to participate. In total, 8 listeners finished the listening test, 5 of them being TTS researchers.

Generally speaking, the annotations can also be obtained from naive listeners. In that case, a larger pool of listeners is needed, and the perceptual relevance threshold thr defined in the following section needs to be increased.

3.2. Listening Test Evaluation

Generally speaking, it is not a simple matter, to evaluate an annotation listening test. One of the concerns always is how to identify non-reliable listeners. This particular issue was not a problem in our study as all participants were highly motivated to provide good quality annotations.

Another issue is the different sensitivity of each participant to various kinds of artifacts. In order to evaluate the perceptual relevance of the outliers, keeping the sensitivity issue in mind, the H_L *score* (2) was introduced:

$$H_L(i) = \frac{\sum_{n=i-L}^{i+L} D_n}{(2L+1) \times N}, \tag{2}$$

where L stands for a tolerance interval length, i is the index of a given outlier, N is the number of listeners. D_n, the number of annotations of a particular phoneme, is defined as follows:

$$D_n = \sum_{i=1}^{N} h_n(j), \tag{3}$$

where $h_n(j)$ is an annotation of the phoneme n defined as:

$$h_n(j) = \begin{cases} 1 & n \in A_j \\ 0 & n \notin A_j \end{cases} \tag{4}$$

where the set A_j is the list of indeces of phonemes annotated by the j-th listener. The H_L *score* in fact represents the number of annotations obtained for each phoneme and its close neighborhood.

Having the H_L *score* defined, each position of a unit of an outlying cost or sub-cost (hereafter referred to as "outlier") was assigned its value. Fig. 2-3 show all outliers and their H_L *scores* sorted by groups corresponding to the concatenation cost sub-components and the costs themselves. Note that the length of the tolerance interval was set to $L = 2$, which was motivated by the above mentioned observation that most listeners used 3–5 phoneme long segments for annotating.

To further quantify the perceptual relevance of the outliers, we have defined a perceptual threshold[1] $thr = 0.5$ for the sum of H_{0-2} *scores* of a particular phoneme (hereafter referred to as $S_2(i)$), and calculated hit/false alarm rates. Summing the H_L *scores* up to the length L allows for normalizing the relevance of artifacts annotated exactly at a particular phoneme with those annotated less precisely. The *Hit Rate* was defined as:

$$Hit\ Rate = \frac{N_{hit}}{N_{outl}} \times 100\,[\%], \tag{5}$$

[1]Experiments showing the impact of different settings of the perceptual threshold thr are presented in [13]. Based on those experiments, the value $thr = 0.5$ used in the current paper was set for the sake of clarity of the method explanation.

Table 1: *The perceptual relevance of the outliers of the concatenation cost sub-components.*

	En	F0	MFCC
Hit Rate [%]	31.25	80.00	45.45
False Alarms [%]	68.75	20.00	54.55
Missed Rate [%]	82.98	91.49	89.36

Table 2: *The perceptual relevance of the outliers of the unit selection costs.*

	Join Cost	TgtCost	TotCost
Hit Rate [%]	83.33	40.79	33.33
False Alarms [%]	16.67	59.21	66.67
Missed Rate [%]	89.36	59.57	91.49

where N_{hit} is a number of outliers of a given cost or a cost sub-component for which the condition $S_2(i) \geq thr$ is fulfilled, and N_{outl} stands for a number of all outliers found for a given cost or cost sub-component. Analogically, the the *Missed Rate* can be defined as:

$$Missed\ Rate = \frac{N_{mis}}{N_{annot}} \times 100\,[\%]\,, \qquad (6)$$

where N_{mis} is a number of *annotated artifacts*, i.e. phonemes fulfilling the condition $S_2(i) \geq thr$, that do not match any outlier position, and N_{annot} is the total number of annotated artifacts.

The results are summarized in Tab. 1-2. We also present the percentage of the annotated audible artifacts missed by each of the costs and their sub-components. In total, 36.17% of the annotated artifacts are not identified by either of the outliers. It is interesting to compare for example the results obtained for the $F0$ and $MFCC$ sub-components of the concatenation cost. It can be seen that both sub-components miss about the same number of *annotated artifacts*, but the $F0$ sub-component shows considerably higher *Hit Rate*.

4. Perceptual Preference Experiment

Having the results of the annotation experiment, it was interesting to speculate how the quality of the synthetic utterances change (if at all) when a limit is set on the costs and their sub-components during searching for the optimal sequences of units forming the test sentences. In other words, what impact has pruning of the search beam based on a pre-set maximum allowed value for the costs and their sub-components.

Obviously, too radical pruning of the search space can lead to inability of the search algorithm to deliver the target sequence of phonemes. Nevertheless, having a large unit database on hand, such an experiment can be conducted. Each concatenation cost sub-component, as well as the costs themselves, were assigned a maximum threshold equal to the value of the upper whiskers of the respective boxplots shown in Fig. 1 (note that the whiskers are placed using 1.5 times the interquartile range; more details can again be found in [13]).

Table 3: *The impact of setting a limit on the concatenation cost sub-components.*

	En	F0	MFCC
Improvement [%]	31.25	41.67	33.33
Deterioration [%]	18.75	16.67	16.67
No impact [%]	50.00	41.67	50.00

Table 4: *The impact of setting a limit on the unit selection costs.*

	Join Cost	TgtCost	TotCost
Improvement [%]	50.00	66.67	66.67
Deterioration [%]	10.00	0.00	33.33
No impact [%]	40.00	33.33	0.00

To evaluate the impact of the modification of the search algorithm, the ABX preference test was conducted. The test sentences were re-synthesized using the modified system and presented to listeners in randomized pairs together with their original versions. The test participants were the same as in the annotation listening test. The task of the listeners was to express their preference regarding the overall quality of the samples. The test also contained sentences that were identical due to not containing any outliers. These sentences were used to check the reliability of the ratings as no preference was expected for the pairs containing them. Again, no visual information was provided to the listeners.

The following results were obtained: 5-prefer original, 9-no preference and 10-prefer modified version. The figures represent ratings for which 60% of listeners found an agreement, also the pairs containing the identical sentences are not included.

The obtained results show a slight preference to the modified system. Despite the fact that the preference is clearly not statistically significant, it is still interesting to analyze removal of which outliers lead to the largest improvement rate. The result of this analysis is shown in tables Tab. 3-4 and will be discussed in the section to follow.

5. Discussion

Let us first take a look at the results obtained for the target cost. It can be seen that removing the related outliers seems to lead to improvements of the system. This is in contrast to the perceptual importance of the target cost outliers obtained in the first test. We believe that this discrepancy is due to the different nature of the two perceptual experiments. While the first one poses implicitly the requirement on the listeners to mark as short segments as possible, the target cost would actually require the opposite as it is rather a supra-segmental cost. On the other hand, setting a limit on the target cost has bigger effect

on the behavior of our system. This is because the target cost outliers appear in larger quantities due to to a large extend binary nature of the cost, and also because when the target cost is "violated", our system stays with this "violation" as long as the concatenations are believed to be smooth according to the concatenation cost or not needed at all.

If we turn next to the concatenation cost and its sub-components, it can be seen that a better consistency was found between the two experiments. In line with the discussion in the previous paragraph, this is perfectly understandable result. Also, it comes as no surprise that $F0$ is the most important sub-component of the concatenation cost in our current implementation. This observation is supported by previous experiments showing fine-grained $F0$ contours as powerful predictors of concatenation discontinuities [14].

Finally, setting the limits on the costs and their sub-components is only one of the potentially possible ways of avoiding units of outlying costs in the selected sequences of units. An interesting alternative could be to tune the unit selection weights using zero number of units with outlying costs as a tuning objective.

6. Conclusions and Future Work

In this paper, two perceptual experiments were presented aiming at the analysis of the audible artifacts present in synthetic speech produced by the unit selection based system. The first experiment forms together with the detection of the outlying values of the unit selection costs and their sub-components a powerful analytic method for the unit selection based TTS systems. The method is driven by the actual costs of an evaluated system, which allows for leveraging the method for the analysis of different systems. We would like to invite interested readers to cooperate on measuring their unit selection implementations using the proposed method.

It has been found that only marginal system improvement can be achieved for our system by setting a limit on the costs during search for the optimal sequences of units. This can be due to data scarcity in our acoustic database. The bigger concern however is the rather low perceptual relevance of the currently used costs and their sub-components. In order to achieve a bigger quality improvement, a rigorous analysis of perceptual cues have to be undertaken. The need for this analysis is further amplified by the observation that 36.17% of the artifacts annotated by the listeners remain unindetified by either of the currently used costs and their sub-components. At the same time, a large number of units with outlying costs do not correspond to audible artifacts annotated by the listeners.

We plan to further experiment with the proposed approach by its extension into more voices and larger sets of data. It is also our intention to conduct the experiment proposed in the last paragraph of the previous section, i.e. to tune our system against the criterion of minimum number of outlying units.

Finally, we also want to look more closely at the audible artifacts that do not correspond to extreme values in the currently used costs and investigate whether or not they can be due to mislabelings in our acoustic database.

7. Acknowledgements

Support for this work was provided by the TA CR, project No. TA01011264 and by the European Regional Development Fund (ERDF), project "New Technologies for the Information Society" (NTIS), European Centre of Excellence, ED1.1.00/02.0090.

8. References

[1] A. Hunt and A. Black, "Unit selection in a concatenative speech synthesis system using a large speech database," in *ICASSP '96*, vol. 1, Atlanta, Georgia, May 1996, pp. 373–376.

[2] E. Klabbers and R. Veldhuis, "On the reduction of concatenation artefacts in diphone synthesis," in *ICSLP '98*, Sydney, Australia, 1998, pp. 1983–1986.

[3] J. Vepa, "Join cost for unit selection speech synthesis," Ph.D. dissertation, University of Edinburgh, 2004.

[4] J.-D. Chen and N. Campbell, "Objective distance measures for assessing concatantive speech synthesis," in *EUROSPEECH '99*, Budapest, Hungary, September 1999, pp. 611–614.

[5] D. Tihelka, J. Kala, and J. Matoušek, "Enhancements of Viterbi search for fast unit selection synthesis," in *INTERSPEECH '10*, Makuhari, Japan, 2010, pp. 174–177.

[6] S. Sakai, T. Kawahara, and S. Nakamura, "Admissible stopping in Viterbi beam search for unit selection in concatenative speech synthesis," in *ICASSP '08*, Las Vegas, USA, 2008, pp. 4613–4616.

[7] J. Matoušek, D. Tihelka, and J. Romportl, "Current state of Czech text–to–speech system ARTIC," in *Proc. of the 9th International Conference TSD 2006, Lecture Notes in Artificial Intellingence*, vol. 4188. Springer Berlin / Heidelberg, 2006, pp. 439–446.

[8] Z. Hanzlíček, "Czech HMM-based speech synthesis," in *Proc. of the 13th International Conference TSD 2010, Lecture Notes in Artificial Intelligence*, vol. 6231. Germany: Springer Berlin / Heidelberg, 2010, pp. 291–298.

[9] D. Tihelka, "Symbolic prosody driven unit selection for highly natural synthetic speech," in *INTERSPEECH '05*, Lisbon, Portugal, 2005, pp. 2525–2528.

[10] D. Tihelka and J. Matoušek, "Unit selection and its relation to symbolic prosody: A new approach," in *INTERSPEECH '06*, vol. 1, Pittsburgh PA, USA, 2006, pp. 2042–2045.

[11] J. Romportl and J. Matoušek, "Formal prosodic structures and their application in NLP," in *Proc. of the 8th International Conference TSD 2005, Lecture Notes in Artificial Intelligence*, vol. 3658. Germany: Springer Berlin / Heidelberg, 2005, pp. 371–378.

[12] M. Legát, "Impact of phonetic context mismatches on quality of vowel concatenations," in *Proceedings of 2012 IEEE 11th International Conference on Signal Processing*, Beijing, China, October 2012, pp. 523–526.

[13] ——, "Configuring TTS evaluation method based on unit cost outliers detection," in *Proc. of the 16th International Conference TSD 2013, Lecture Notes in Artificial Intelligence*, 2013, p. (accepted).

[14] M. Legát and J. Matoušek, "Pitch contours as predictors of audible concatenation artifacts," in *Proceedings of the World Congress on Engineering and Computer Science 2011*, San Francisco, USA, 2011, pp. 525–529.

Development of Electrolarynx with Hands-Free Prosody Control

Kenji Matsui[1], *Kenta Kimura*[1], *Yoshihisa Nakatoh*[2], *Yumiko O. Kato*[3]

[1] Osaka Institute of Technology, Osaka, Japan
[2] Kyushu Institute of Technology, Kitakyushu, Japan
[3] St. Marianna University School of Medicine, Kawasaki, Japan

matsui@elc.oit.ac.jp

Abstract

The feasibility of using a motion sensor to replace a conventional electrolarynx(EL) user interface was explored. Forearm motion signals from MEMS accelerometer was used to provide on/off and pitch frequency control. The vibration device was placed against the throat using support bandage. Very small battery operated ARM-based control unit was developed and placed on the wrist. The control unit has a function to convert the tilt angle into the pitch frequency, as well as the device enable/disable function and pitch range adjustment function. As for the forearm tilt angle to pitch frequency conversion, two different conversion methods, linear mapping method and F0 template-based method, were investigated. A perceptual evaluation, with two well-trained normal speakers and ten subjects, was performed. Results of the evaluation study showed that both methods were able to produce better speech quality in terms of the naturalness.

Index Terms: prosody, electrolarynx, hands-free

1. Introduction and Related work

People who have had laryngectomies have several options for the restoration of speech, but no currently available device is satisfactory. The artificial larynx, typically a hand-held device which introduces a source vibration into the vocal tract by vibrating the external walls, is the easiest for patients to master, but does not produce airflow, so the intelligibility of consonants is diminished and the speech is uttered at a monotone frequency. Alternatively, esophageal speech does not require any special equipment, but requires speakers to insufflate, or inject air into the esophagus, and limits the pitch range and intensity. Both esophageal speech and tracheo-esophageal speech are characterized by low average pitch frequency, large cycle-to-cycle perturbations in pitch frequencies, and low average intensity. As for utilizing esophageal speech, it was found that age was the important factor. When laryngectomized patients get older, they face difficulty in mastering the esophageal speech or keep using esophageal speech because of the waning strength. For that reason, the electrolarynx is an important device even for the people who use esophageal speech.

As for the advantages of EL, firstly, one can speak in long sentences that are easily understood. Secondly, no special care requirements are needed; the EL only has to be placed up against the neck and turned on. Thirdly, the EL can be used by almost everybody, regardless of the post-operative changes in the neck. In those few cases where scarring prevents proper placement of the EL, an intraoral version can be used.

On the other hand, there are a couple of disadvantages. Firstly, the EL has a very mechanical tone that does not sound natural. There usually is little change in pitch or modulation. Secondly, one must use their hand to control the EL all the time, and its appearance is far from normal.

Pitch frequency control is one of the important mechanisms for EL users to be able to generate naturally sounding speech. There are some commercially available EL devices using a single push button with pressure sensor to produce F0-contours [1], [2]. There are also similar studies of pitch controlling methods [3], [4]. However, none are hands-free. The approaches for generating F0-contour without manual interaction have been proposed. Saikachi et al. use the amplitude variation of EL speech [5]. Another approach is to generate a F0-contour using an air-pressure sensor that is put on the stoma [6], [7]. Also, recently, a machine learning F0-contour generation from the EL speech has been proposed [8]. Although most of those studies are in an early stage of research, the results show substantial improvement of EL speech quality.

An EL system that has a hands-free user interface could be useful for enhancing communication by alaryngeal talkers. Also, the appearance can be almost normal because users do not need to hold the transducer by hand against the neck. Almost all people frequently use gestures when they talk. It would be quite convenient if the EL users could utilize gestures to control the device. Furthermore, gesture control has a lot of potential to handle not only just on/off function, but also many other functions because hands can generate various types of motion. However, if users can not even use hand gestures, for example, controlling something, we need to consider other part of body movement, or completely different technique, such as EMG based hands-free EL control [9].

The present study was undertaken to explore the feasibility of using gesture control method to replace the conventional EL user interface in terms of both on/off function and pitch control. Also, a wrist-watch type EL control device was designed and evaluated in order to determine the actual speech generation performance in a real environment. The specific goals were: 1) to determine the practical hands-free user interface method for EL system, and 2) to determine whether the generated speech has high intelligibility and naturalness.

2. Determination of Needs – Survey Results

2.1. User Profile

A set of techniques — including user observations, interviews, and questionnaires — were used to understand implicit user needs. As for the questionnaire survey, the total number of laryngectomized participants was 121 (87% male, 13% female), including 65% esophageal talkers, 12% EL users, 7% both, and 21% used writing messages to communicate.

2.2. Survey Results

Almost all of the participants claimed that most public areas are difficult for oral communication due to the noisy environment. Typical public areas include train stations, inside of train cars, inside of vehicles, restaurants/pubs, and conventions/gatherings.
The noisy environment issue is well known problem and people usually use portable amplifier, however, we have been investigating smaller, lighter, and low profile speech enhancement system for both esophageal speech [10] and EL.
Other needs confirmed from the survey are:

- Naturally sounding voice, not like mechanical tone
- Light weight device
- Smaller device, low profile
- Hands-free, easy to use
- Low cost

Based on those survey results, present study was conducted to meet the essential user needs.

3. Hands Free UI Design

3.1. Gesture Control

Gesture control UI can be developed through the use of a system based on photo detector, camera, or accelerometer. Based on the survey results, a three-axis MEMS accelerometer was used in this study. MEMS sensors are very small, low cost, and fit the system requirements well [12].

3.2. Pitch Control

A MEMS accelerometer accurately measures acceleration, tilt, shock and vibration in applications. The challenge in designing the pitch control algorithm that use a MEMS accelerometer output to control pitch contour is to reconcile the numerical ranges between two types of data. MEMS output bytes are integers in the range -128 to 127 for a range of ±2G. Often this issue can be easily reconciled by linear mapping of one range of values (such as MEMS data values -128 to 127) into another range (such as 67 to 205 expected as the typical male pitch range).

Another possible pitch control method is to utilize a pitch contour generation model, such as Fujisaki's model [11]. The system needs to have a strategy to generate both the phrase component and the accent component from the MEMS output. The F0 template-based method is easier to generate relatively stable pitch contour, however, it may lose some flexibility to generate various pitch patterns.

In this study, both the simple linear mapping method and the F0 template-based method were prototyped and examined to evaluate the pitch control performance. Also, the comparison study was performed between conventional EL, the linear mapping method and the F0 template-based method.

4. System Implementation

4.1. Hardware System Design

The pitch control algorithms described above were implemented on a small CPU board. A block diagram of the hardware architecture is shown in Fig. 1. A pair of very small EL transducer with neck-bandage has also been prepared to place it to the optimal location on the neck. We introduced the small hardware in order to meet the user requirements, i.e. small, comfortable weight, and low cost. The ARM-based hardware unit consists of a small board (34mm×34mm) with a 48MHz C1114, a 32 bit ARM cortex-M0, a 10 bit PWM with 10kHz sampling rate, a USB interface, 32kB FLASH memory, and three 1.5V batteries. Picture 1 shows the ARM unit and the EL transducers.

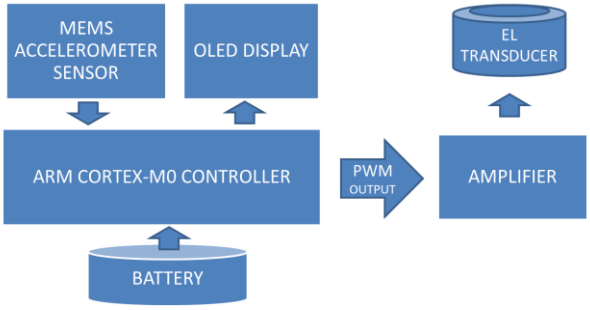
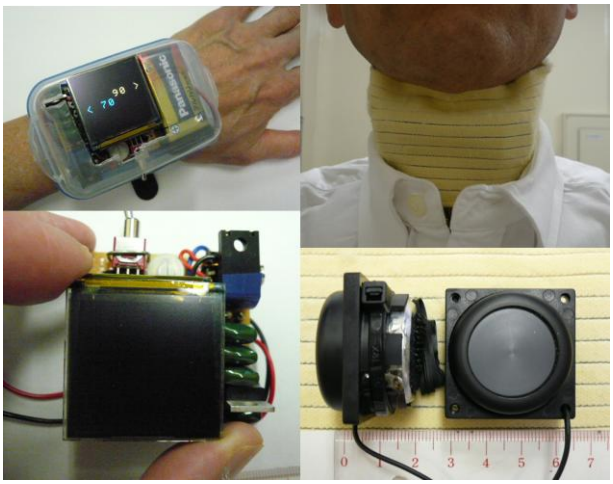


Figure 1: *Block diagram of the Hardware Architecture*



Picture 1: *EL Controller and Transducer unit (upper left: EL controller on the wrist, lower left: ARM-PC board, upper right: transducer with neck bandage, lower right: transducer unit)*

4.2. Pitch Control (linear mapping method)

Hand gestures are a very important part of language. A preliminary UI study using forearm movement was conducted in order to evaluate feasibility of the pitch control mechanism. Fig.2 shows the forearm tilt and the MEMS output (x-axis) when the controller was placed on the wrist. From the horizontal position (0°) to the 75°upward position is the normal pitch control zone. From the horizontal position to the -25° downward position is the fading out zone, where phrase ending pitch pattern is adjusted based on the forearm moving speed. As for the conversion from the MEMS output to the pitch frequency, there are four pitch ranges. Fig.3 shows the relation between the MEMS output and the four ranges of pitch frequency, i.e. high, mid-high, mid-low, and low. Users can select one of the four ranges.

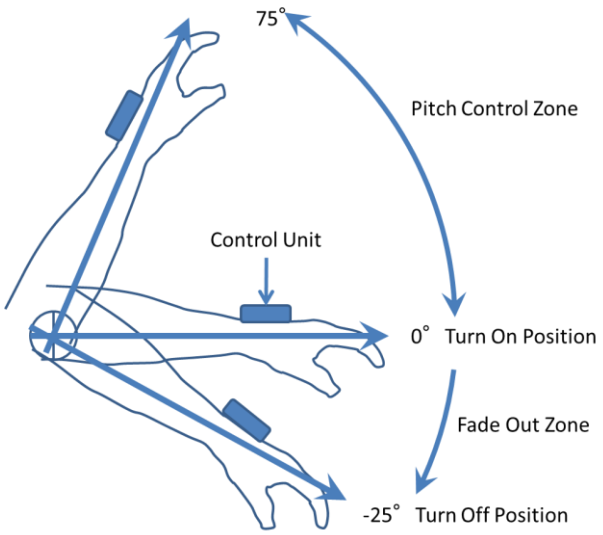


Figure 2: *Forearm Tilt and Pitch Control*

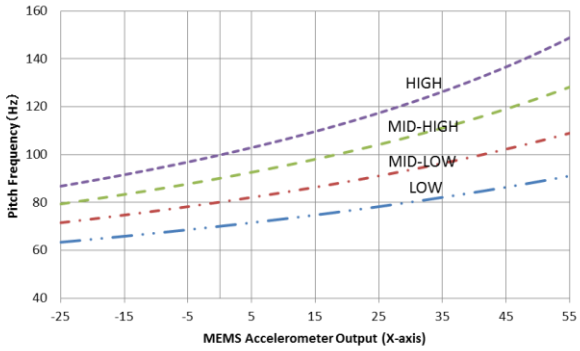


Figure 3: *Relation between MEMS Output and Pitch (linear mapping method)*

4.3. Pitch Control (F0 tmplate-based method)

The linear mapping method is straight forward approach, however, it requires precise sensor control to avoid unnatural pitch behavior. The F0 template-based method applies a basic F0 template to the fine F0 contour generation. The phrase component of Fujisaki's model was used to generate the F0 template *F0*(*t*). While the system is intended to generate both phrase control and accent control, as the first step of testing the template, we utilized only the phrase component.

$$\ln F0(t) = \ln F\min + Ap \cdot Gp(t) \quad (1)$$

where

$$Gp(t) = \alpha^2 t \exp(-\alpha t) \quad (2)$$

The symbols in equations(1) and (2) indicate:

- *F*min is the minimum value of speaker's F0.
- *Ap* is the magnitude of phrase command.
- α is natural angular frequency of the phrase control mechanism.

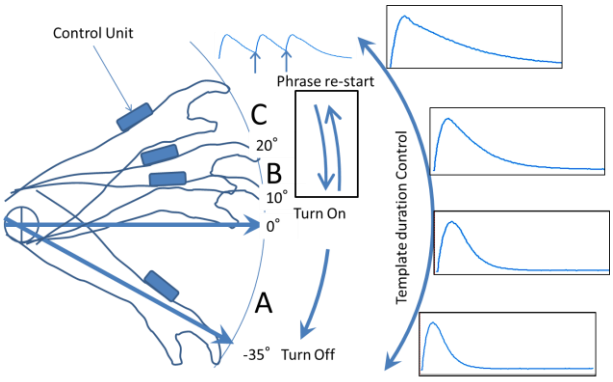


Figure 4: *Relation between Forearm tilt and F0 template generation (F0 model-based Method) A-zone: -35°~0°, B-zone: 0°~20°, and C-zone: 20°~.*

In this study, those values are; *F*min = 80Hz, α =1.5, and *Ap*=0.75. The calculated F0 template data is stored in the controller software. Fig.4 shows the F0 contour generation mechanism using the MEMS sensor output and the F0 template. The oscillation starts at 10° upward from the horizontal position. The template duration is controlled based on the forearm tilt angle as shown in the Figure 4. Also, in that figure, how to re-start the F0 template is shown. Basically, the forearm movement (C-zone → B-zone → C-zone) is required. A-zone is -35°~0°, B-zone is 0°~20°, and C-zone is 20°~, respectively.

4.4. ON/OFF Control

Reliable EL ON/OFF control is very important for users to talk comfortably. As you can see in Fig.2, EL vibrates at the normal pitch control zone, i.e. from 0° position and higher. EL stops the vibration at the -25° position or lower. The hysteresis is necessary to avoid unstable behavior near the on/off threshold. If the phrase does not have an accent, the pitch rises from a low starting point on the first mora, and then levels out. Such pitch contour is generated by moving the forearm downward very quickly. However, most of the accented phrases are generated by gradual movement. In case of the F0 template-based approach, in order to turn on the EL device, the tilt angle needs to be 10° or higher, and the turning off position is at -35°.

4.5. LOCK Mechanism

It is very important to enable/disable the controller easily and quickly while users are wearing the device. Y-axis output of the MEMS accelerometer was used to implement such a lock mechanism. By twisting the wrist quickly and generating 2G acceleration, the user can enable/disable the EL.

5. Perceptual Evaluation

Subjective evaluation tests (by rating scale method) have been made with 2 male well-trained normal speakers, and 10 (one female and nine male) subjects. Each speaker read the phonetically balanced test materials as shown in table 1. We used one commercially available EL device (SECOM EL-X0010), prototype-A (linear mapping method, with 70Hz mode), and prototype-B (F0 template-based method). Those 60 speech stimuli (2speakers ∗3devices∗10sentences) were recorded, and two sets of differently randomized stimuli were

prepared. 5 subjects evaluated the one set of stimuli, and other 5 subjects rated the other set of stimuli. Each speech stimuli were presented two times.

Table 1: *Phonetically balanced Japanese test sentences*

1	Papa mo mama mo minna de mamemaki o shita.
2	takai takai tokoro e nobotte iku tokoro da.
3	Achirakara mo kochirakara mo dochirakara mo ikukoto ga dekiru.
4	Aoi o ueru.
5	Anohito wa bunkajin to yobareru no ga fusawashî.
6	Shichi gatsu kara hanshin densha de tûkin shite ima su.
7	Ginkô mo gakkô mo aruite ikeru kyori ni ari masu.
8	Kinkô ga tore te iru no de kakkô ga yoi.
9	shijû gamu o kamuno ga syûkan ni natte iru.
10	Hana o ottari ana o hottari sanzanna meni atta.

The subjects rated the speech stimuli in terms of "intelligibility (Clarity)", "naturalness of the prosody", and "stability of the prosody" using five level scaling. As shown in Figure 5-(a), 5-(b) and 5-(c), the subjective evaluation indicated that both prototype-A(LM) and B(FU) obtained higher naturalness scores than the EL device(EL). On the other hand, intelligibility(clarity) and stability shows almost no difference among those devices.

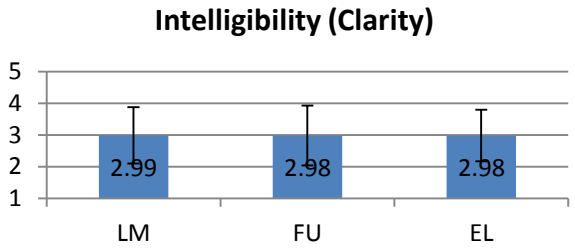


Figure 2-(a): average evaluation scores of intelligibility

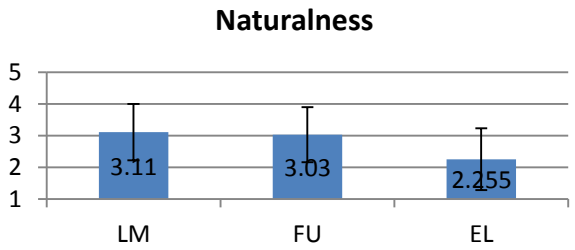


Figure 5-(b): average evaluation scores of naturalness

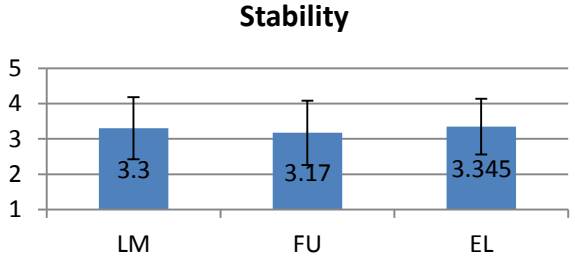


Figure 5-(c): average evaluation scores of "Stability"

6. Discussion

Without losing intelligibility(clarity) and stability of the prosody, both prototype-A and B showed substantial improvement in terms of the naturalness of the prosody. The results of this study indicate that both usability and speech quality of EL speakers could be improved by MEMS accelerometer based hands-free UI controller. The ability to control the pitch contour of EL speech with the proposed linear mapping method and F0 template-based method implies that hand gesture control may be adequate for implementation of the hands free user interface for EL device. Our assumption about the performance difference between the two proposed methods is that the F0 template-based method may be easier to learn and easier to stabilize the pitch contour, however, there was almost no difference between those two methods. We plan to run the same evaluation with actual EL-users, and confirm if the proposed methods show similar performance. Also, a more detailed and precise study across the talkers, sentences, and learning curve has to be performed. As for the gesture control, we tested only the forearm movement, however, it is necessary to test other body locations where users might be able to control the EL device more easily and naturally. According to the user requirements, the evaluation of appearance also needs to be considered. In the study, we set a relatively narrow pitch range in order to avoid wild swings in pitch. A better pitch control range needs to be investigated.

7. Conclusion

MEMS accelerometer based hands free UI for EL device was proposed. A hand gesture control unit was designed and prototyped. Two types of pitch contour generation methods were proposed and tested together with conventional EL device. Results of the evaluation indicated that the proposed methods have a potential to make the EL output prosody more natural, easy to use, and less distinct appearance. However, a more detailed and precise study across the talkers, sentences, and learning curve has to be performed.

8. Acknowledgements

This work was supported by JSPS KAKENHI Grant-in-Aid for Scientific Research(C) Grant Number 24500664.

9. References

[1] SECOM company Ltd., Electrolarynx "MY VOICE", (http://www.secom.co.jp/personal/medical/myvoice.html)
[2] Griffin laboratories, TruTone users guide.
[3] Y. Kikuchi, and H. Kasuya, : "Development and evaluation of pitch adjustable electrolarynx", In SP-2004, 761-764, 2004
[4] H. Takahashi, M. Nakao, T. Ohkusa, Y. Hatamura, Y. Kikuchi, and K. Kaga, 2001. Pitch control with finger pressure for electrolaryngial or intra-mouth vibrating speech.Jp. J. Logopedics and Phoniatrics, 42(1),1-8.
[5] Y. Saikachi, "Development and Perceptual Evaluation of Amplitude-Based F0 Control in Electrolarynx Speech", Journal of Speech, Language, and Hearing Research Vol.52 1360-1369 October 2009
[6] N. Uemi, T. Ifukube, M. Takahashi and J. Matsushima, "Design of a new electrolarynx having a pitch control function", In Proceedings of 3rd IEEE International Workshop on Robot and Human Communication, RO-MAN p.198-203, Nagoya, Japan, July 18-20, 1994.
[7] K. Nakamura, T. Toda, H. Saruwatari and K. Shikano, "The use of air-pressure sensor in electrolaryngeal speech enhancement",

INTERSPEECH, p.1628-1631, Makuhari, Japan, Sept 26-30, 2010.

[8] A. K. Fuchs and M. Hagmüller, "Learning an Artificial F0-Contour for ALT Speech", INTERSPEECH, Portland, Oregon, Sept. 9-13, 2012.

[9] H.L. Kubert, "Electromyographic control of a hands-free electrolarynx using neck strap muscles", J Commun Disord. 2009 May-Jun;42(3):211-25

[10] K. Matsui, et al., "Enhancement of Esophageal Speech using Formant Synthesis", Journal of Acoustical Society of Japan (E) 23,2 pp.66-79, 2002

[11] H. Fujisaki, In Vocal Physiology: Voice Production, Mechanisms and Functions, Raven Press, 1988

[12] K. Matsui, et al., "A preliminary user interface study of speech enhancement system", Proc. of the 1st International Conference on Industrial Application Engineering 2013, 53-56

A Hybrid TTS between Unit Selection and HMM-based TTS under limited data conditions

Trung-Nghia Phung[1,2], *Chi Mai Luong*[2], *Masato Akagi*[1]

[1]Japan Advanced Institute of Science and Technology, Ishikawa, Japan
[2]Institute of Information Technology, Hanoi, Vietnam

ptnghia@jaist.ac.jp, lcmai@ioit.ac.vn, akagi@jaist.ac.jp

Abstract

The intelligibility of HMM-based TTS can reach that of the original speech. However, HMM-based TTS is far from natural. On the contrary, unit selection TTS is the most-natural sounding TTS currently. However, its intelligibility and naturalness on segmental duration and timing are not stable. Additionally, unit selection needs to store a huge amount of data for concatenation. Recently, hybrid approaches between these two TTS, i.e. the HMM trajectory tiling TTS (HTT), have been studied to take advantages of both unit selection and HMM-based TTS. However, such methods still require a huge amount of data for rendering. In this paper, a hybrid TTS among unit selection, HMM-based TTS, and the Modified Restricted Temporal Decomposition (MRTD), named HTD, is proposed motivating to take advantages of both unit selection and HMM-based TTS under limited data conditions. Here, TD is a sparse representation of speech that decomposes a spectral or prosodic sequence into two mutually independent components: static event targets and correspondent dynamic event functions, and MRTD is a compact but efficient version of TD. Previous studies show that the dynamic event functions of MRTD are related to the perception of speech intelligibility, one core linguistic or content information, while the static event targets of MRTD convey non-linguistic or style information. Therefore, by borrowing the concepts of unit selection to render the event targets of the spectral sequence, and directly borrowing the prosodic sequences and the dynamic event functions of the spectral sequence generated by HMM-based TTS, the naturalness and the intelligibility of the proposed HTD can reach the naturalness of unit selection, and the intelligibility of HMM-based TTS, respectively. Due to the smoothness of event functions of MRTD, an appropriate smoothness in synthesized speech can still be ensured when being rendering by a small amount of data, resulting in the usability of the proposed HTD under limited data conditions. The experimental results with a small Vietnamese dataset, simulated to be a "limited data condition", show that the proposed HTD outperformed all HMM-based TTS, unit selection, HTT under a limited data condition.

Index Terms: TTS, unit selection, HMM-based, Temporal Decomposition, HTT

1. Introduction

Building a large-scale speech corpus is a costly task that takes a long time and a great deal effort by engineers, acousticians and linguists. Therefore, to build high-quality speech synthesizers under limited data conditions is important in practice, specifically for under-resourced languages.

The two most successful TTS up to now are unit selection [1] and HMM-based [2, 3]. HMM-based TTS provides synthetic speech with stable and smooth trajectory. Therefore, the intelligibility of HMM-based TTS can reach that of the original speech. However, the naturalness of HMM-based TTS is low mainly due to the spectral over-smoothness caused by the "averagely" statistical processing. Although much research has been attempted to reduce this over-smoothness [3], speech synthesized by HMM-based TTS is still muffled and far from natural. On the contrary, the naturalness of unit-selection TTS is high. However, unit-selection has some drawbacks reducing the range of its practical applications. Among them, one main drawback is the unstability of the temporal trajectory of speech synthesized by unit selection. This artifact reduces the intelligibility of the speech synthesized by unit selection TTS compared with that of HMM-based TTS [4]. Another main drawback of unit selection is the requirement of a huge data corpus for concatenation.

Recently, hybrid approaches, which use HMM to guide the unit selection process to improve the stability and the smoothness of unit selection TTS, have been shown as the most successful TTS that can preserve the advantages of both HMM-based and unit selection TTS. Among them, the HTT [4] can be considered as the state-of-the-art hybrid TTS. In this system, HMM trajectory is used to guide the selection of each 5ms frame to concatenate the waveforms. The naturalness of this hybrid TTS is comparative with that of state-of-the-art unit selection TTS, while the intelligibility of this hybrid TTS is comparative as that of state-of-the-art HMM-based TTS. However, HTT [4] still requires a huge amount of data for rendering due to the use of "frame selection". If the selection process is imperfect due to some reasons such as the limitation of the data corpus, it is easy to perceive the discontinuities between frames. The experimental results in [4] show that the quality of HTT is stable only if the amount of stored database for rendering is approximately 2 hours to 10 hours.

In this study, in order to borrow the high naturalness of unit selection and the high intelligibility of HMM-based TTS, MRTD [6] is used to decompose the spectral sequence generated by HMM into two mutually independent components: static event targets and correspondent dynamic event functions, in which one component is related to the perception of intelligibility and the other one is related to the perception of naturalness. These two components are independently controlled in order to reach the intelligibility of HMM-based TTS and the naturalness of unit selection TTS. The proposed method is supported by previous studies [7, 8], where the temporal event functions can represent the "languageness" or linguistic information of speech, which is important for the perception of speech intelligibility, while the sparse event targets can convey non-linguistic

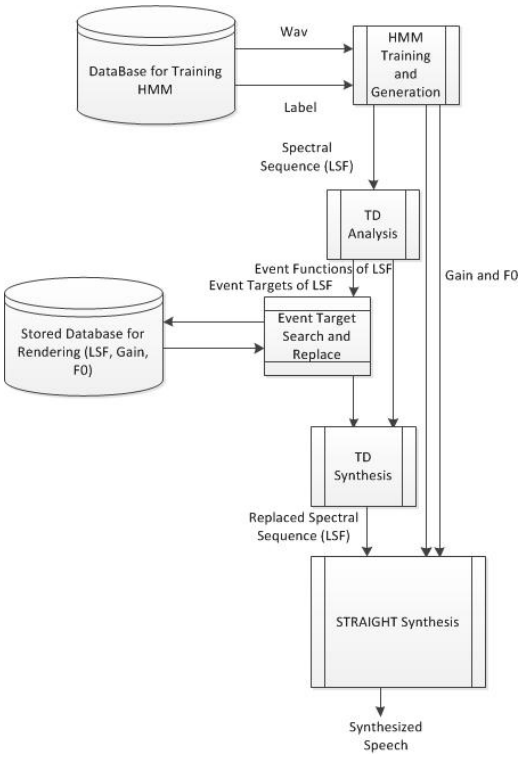


Figure 1: General Diagram.

of style information, which is important for the perception of the speech naturalness [7, 8].

Comparing with HTT [4], the proposed HTD has one main advantage on reducing the required amount of data for rendering, resulting in improved quality of the synthesized speech under limited data conditions. It reveals that HTT replaces all frames of the guided trajectory generated by HMMs with the closest frames found in the original database. Therefore, to ensure the stability and the smoothness of the synthesized speech, HTT requires a huge database for rendering because the limitation of data may cause the mismatches and the discontinuities between the consecutive replaced frames. However, in the proposed HTD, the smoothness of the spectral and prosodic trajectories is ensured by the smoothness of the event functions of the spectral sequence generated by HMM and the smoothness of prosodic sequences generated by HMM. Therefore, the matching level of the "target selection" task is not strictly required as in HTT [4], resulting in the reduction of the required amount of data for rendering.

In the next section, we will present and explain the details of the proposed HTD. Section 3 describes the evaluations, and finally section 4 draws the conclusions.

2. The proposed HTD

2.1. Outline of the proposed TTS

The general diagram of the proposed HTD is shown in Fig. 1.

At the first stage, spectral and prosodic trajectories are generated from HMM-based TTS. Since HMM-based TTS is efficient on prosodic modeling [5], the prosodic trajectories of the F0 contour and gain contour of HMM-based TTS are preserved for the proposed HTD.

At the second stage, the line spectral frequency (LSF) sequence generated by HMM-based TTS is analyzed by TD analysis [9] using a simplified TD version called MRTD [6].

Assume that the $\boldsymbol{y(n)}$ is this spectral sequence, TD decomposes $\boldsymbol{y(n)}$ into dynamic event functions ϕ and K static event targets $\boldsymbol{a}$ among total N frames, as given in Eqs. 1 and 2. Here, $\boldsymbol{\hat{y}(n)}$ is the approximation of $\boldsymbol{y(n)}$. There are K event targets in a total of N frames and $K << N$, then TD is a sparse representation of speech. The event functions are interpolation functions representing temporal transition movements between the sparse event targets. [9].

$$\boldsymbol{\hat{y}(n)} = \sum_{k=1}^{K} \boldsymbol{a_k}\phi_k(n), 1 \leq n \leq N \tag{1}$$

$$\hat{Y}_{P\times N} = \boldsymbol{A_{P\times K}}\Phi_{K\times N} \tag{2}$$

Figure. 2 draws an example of MRTD with spectral parameter $y(1:N)$, event targets $a(1:K)$, and event functions $\phi(1:K)$.

In Eq. (1) (or matrix representation as in Eq. (2)), event target $\boldsymbol{a}$ and event function ϕ are unknown and needed to be estimated by some optimization tasks to minimize interpolation error. In the initiation of the optimization task in MRTD [6], event targets are set equal to the frame-based vector at the same locations as given in Eq. 3.

$$\boldsymbol{a_k} = y(n_k) \tag{3}$$

Here, n_k is the location of event target a_k.

Then, event functions in MRTD are estimated as described in Eqs. (4) and (5). Here, $\langle ... \rangle$ and $||.||$ correspond to the inner product of two vectors and the norm of a vector.

$$\phi_k(n) = \begin{cases} 1 - \phi_{k-1}(n),\ if\ n_{k-1} < n < n_k \\ 1,\ if\ n = n_k \\ \min(\phi_k(n-1), \max(0, \hat{\phi}_k(n))), \\ if\ n_k < n < n_{k+1} \\ 0,\ otherwise \end{cases} \tag{4}$$

$$\hat{\phi}_k(n) = \frac{\langle (y(n) - a_{k+1}), (a_k - a_{k+1}) \rangle}{||a_k - a_{k+1}||^2} \tag{5}$$

Using the estimation given in Eqs. (4) and (5), each event function $\phi_k(n)$ is smooth, has only one peak, and two overlapped event functions sum up to one as described in Fig. 2 and explained in detail in [6]. These properties of event functions results in gradual movements of the interpolated spectral $\boldsymbol{\hat{y}(n)}$ that are related to the co-articulation of speech. In addition, the modification on sparse event targets $\boldsymbol{a_k}$ directly and gradually affects to all frames inside duration in which the event function ϕ_k is non-zero. Hence, speech can be flexibly modified / transformed at specific events in the time domain by modifying / transforming MRTD event targets $\boldsymbol{a}$ as shown in [7, 8].

After estimating event functions, event targets are estimated as in Eq. 6, where ^{T} is matrix transpose transformation.

$$A = Y\Phi^T(\Phi\Phi^T)^{-1} \tag{6}$$

Note that Eq. 6 is the general form of the re-estimation of event targets of LSF in the original MRTD that was described in details in the original work [6]. For short, Eq. 6 means that each event target is re-estimated by its initialized value, which is the frame-based vector at the same location, and the non-zero estimated event functions at the same location with a convergence

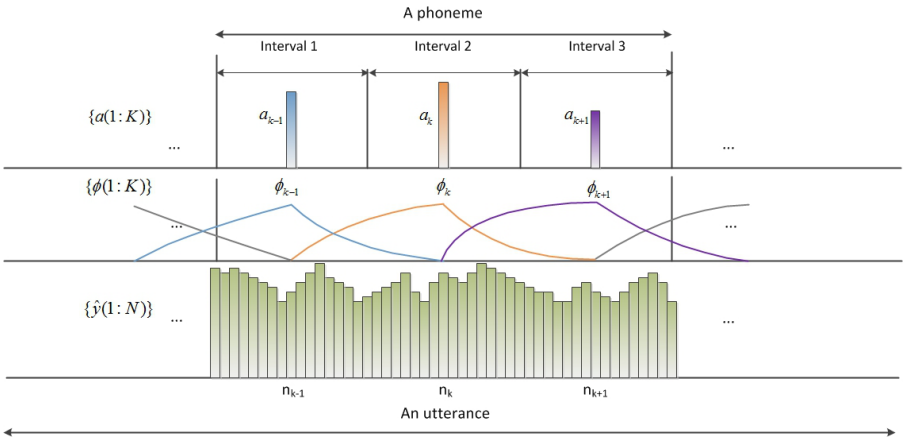


Figure 2: An example of MRTD analysis / synthesis with N frames and K event targets: a bar represents a frame-based spectral feature or a spectral event target in a specific location. Two bars located at two locations of the same spectral event target are same in lengths.

condition of minimizing the reconstruction error and ensuring the orders of LSF.

In the third stage of the proposed HTD, the event functions ϕ_k of the spectral sequence $\boldsymbol{y}$, which are important for speech intelligibility as mentioned in section 1, are also directly borrowed from HMM-based TTS because the intelligibility of HMM-based TTS is stable and high, approximately that of original speech. To overcome the over-smoothness of the spectral sequence generated by HMM and also to transform the spectral sequence to the original speech, the event targets $\boldsymbol{a_k}$ are selected from an original dataset. The target selection procedure is described more detail in the next sub-section.

Finally, at the last stage, the high-quality speech vocoder STRAIGHT [10] is used to generate speech waveforms.

2.2. Event target selection

The proposed method for selection is based on event target, so the proposed TTS can be considered as a new concept "target selection" rather than conventional concepts "unit selection" and "frame selection" [4].

The event targets of the speech trajectory generated by HMM are modified in the proposed HTD by replacing them with the most-matched event targets of original speech. Therefore, an alignment procedure in the time domain is required to accurately modify the event targets of source speech into the corresponding event targets of target speech.

Dynamic time wrapping (DTW) or the nearest neighbor search (NNS) can be used in the frame-based voice transformation to align the transformation in parallel form for the former and in non-parallel form for the latter. A technique of using a fixed number of equally-spaced event targets for each phoneme has been proposed when using TD-based voice transformation [7]. This method involves non-parallel transformation for a syllable or an utterance but is a parallel transformation for each phoneme when each ordered event target of a source phoneme is transformed into a corresponding ordered event target of a target phoneme. Developing from this method, each phoneme is divided into three equally-spaced intervals in this work. One event target is located at the center of each of the three intervals. Therefore, there are three event targets in one phoneme. The number of event targets in one phoneme can be from one as in the original MRTD [6], or five in [7]. There are two reasons for choosing three event targets in one phoneme in this work. The first one is that increasing the number of event targets in one phoneme larger than three does not improve the quality of synthesized speech in our experiments, but increases the size of stored data for rendering. The second one is that we want to set the number of equally-spaced intervals as well as the number of event targets in one phoneme same as the number of HMM states in each phoneme, which is three in this work, with an expectation that all HMM states are rendered by the original data. Although the method of locating event targets at center frames in each HMM state in Viterbi alignment is straightforward and may increase the accuracy of the selection procedure, compared with the use of equally-spaced intervals in the proposed method, this method has not implemented in this research at present. This is one of our future works.

The event target searching and replacing are represented in Fig. 3. Each event target of source spectral sequence generated by HMM is replaced by an event target of the original speech, searched by a selection process. The selection is supervised by labeled data in order to ensure its accuracy and reduce the amount of searching time. Using MRTD analysis, each event target is re-estimated by the frame-based vector at the same location, and the estimated non-zero event functions at the same location, as explained in sub-section 2.1. Therefore, event targets depend on the wide-range context, and sensitive to its locations. As a result, to directly use event targets for alignment may reduce the accuracy of the alignment procedure. Instead of that, three consecutive frames, referred to as tri-frames in this research, located at same positions of event targets, are used to align the source and target event target pairs.

The matched tri-frames are searched by nearest neighbor search with a summed cost as defined in Eqs. 7, 8, 9, 10, and 11.

$$d = N(d_{F0}) + N(d_{LSF}) + N(d_G) \tag{7}$$

$$d_{F0} = |\log(F0_t) - \log(F0_s)| \tag{8}$$

$$d_{LSF} = \sqrt{\frac{1}{P}\sum_{i=1}^{P}(LSF_{i,t} - LSF_{i,s})^2} \tag{9}$$

$$d_G = |\log(G_t) - \log(G_s)| \tag{10}$$

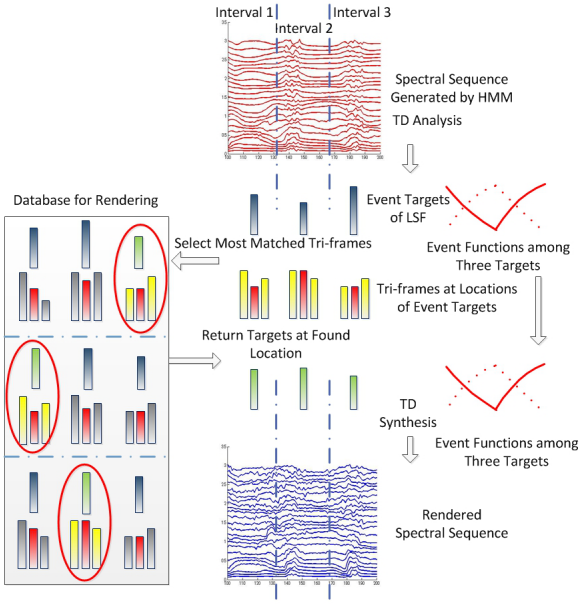


Figure 3: Target Selection: Single bars represent spectral event targets located at centers of equally-spaced intervals; triple bars represent frame-based features in tri-frames where their central frames are located at the same positions as event targets. Input synthetic tri-frames color yellow-red-yellow, selected original tri-frames with the same colors are marked by red circles, and green event targets are the event targets of the original speech selected for replacing.

$$N(d) = \frac{d - \mu_d}{\sigma_d} \tag{11}$$

Here, each cost for LSF, F0, and gain G is computed with a source target pair s and t. Each component cost is normalized by normal distribution similar as in [4], as shown in Eq. 11, where are mean and standard deviation of the sample distances of all candidates, respectively. The ideal behind the use of all F0, LSF, and PL to compute the distance cost between the trajectories generated by HMM-based TTS and those of the original speech in HTT is to find the physically closest frames (in the waveform domain) for concatenation. This ideal was adopted to the proposed HTD to select the spectral target of the physically closest frame in the original database.

After the matched tri-frames have been selected from the original data, the event targets of the spectral sequence generated by HMM-based TTS in the previous stage are replaced by the outputs of the selection procedure, which are the original event targets located in the same positions as the selected tri-frames.

In our implementation, the "target selection" is supervised by labeled data to ensure its accuracy and reduce the length of searching time, in which each ordered target in a phoneme is replaced by the selected targets with the same order and in the same phoneme. In the offline stage, the database for rendering is prepared with two steps. First, all utterances with labels are analyzed by MRTD. Then, analyzed event targets and tri-frames at the same locations are extracted from the parameters of the whole utterances by using label data, and stored for each distinct phoneme. In the online rendering stage for each phoneme, the matched original tri-frames are selected from the original data and the event targets of the spectral sequence generated by HMM-based TTS in the previous stage are replaced by the original event targets located in the same positions as the selected original tri-frames. The "target selection" will be run with the whole database if the target phoneme for rendering is not found. Therefore, the selection procedure can still work if the number of phonemes in the database for rendering is not sufficient, such as under some limited data conditions, for instance.

3. Implementations and Evaluations

3.1. Data preperation

The main motivation of this study is to propose an efficient hybrid TTS under limited data conditions. The TTS under limited data conditions is more practical for under-resourced languages, where huge public speech corpus is missing, compared with high-resourced languages. Vietnamese is a language spoken by about 100 million people in the world. However, there is no huge public speech corpus with labeling for Vietnamese at present. Therefore, Vietnamese is one of the under-resourced languages and a Vietnamese corpus is used in this study.

Vietnamese is a tonal monosyllabic language. There are 20 consonants and 250 tonal vowels in Vietnamese. More detail on Vietnamese can be found in [11]. In this research, we used the small Vietnamese corpus DEMEN567, including 567 utterances. This corpus was also called TTSCorpus in [12]. The total time interval of this dataset is approximately one hour. The sampling frequency of the corpus is 11025 Hz. We simulated a limited data condition by a dataset of 300 utterances. This dataset is close to the threshold where the phoneme coverage reachs approximately 100%. Although some monophones are still mising, most of widely used tonal phonemes appear in this dataset. The size of this dataset in PCM 16 bits format is approximately 30MBs and the duration is approximately 20 minutes.

3.2. Experimental Parameters

We compared the six versions of speech in our evaluations: speech synthesized by a HMM-based TTS for Vietnamese [13], speech synthesized by a non-uniform unit selection TTS for Vietnamese [14], speech synthesized by HTT, speech synthesized by our proposed HTD, speech analyzed / synthesized by MRTD-STRAIGHT, and the original speech. All speech synthesizers used the same dataset simulated to be "under limited data conditions".

Speech analyzed / synthesized by MRTD and STRAIGHT can be considered as the ideal limitation of HTD obtained when using a huge amount of data for rendering. Due to reconstruction errors of MRTD and STRAIGHT, this ideal limitation of HTD is different from the original speech. The original speech can be considered as the ideal limitation of unit selection TTS and HTT when using a huge amount of data for selection or rendering since these synthesizers are waveform concatenation TTSs that use the original speech. Although these two ideal limitations can be never reached, they were used for evaluations in this paper instead of evaluating the synthesizers with a real large-scaled speech corpus because the latter solution is expensive, time-consuming, and not available for us at present.

All experimental parameters were controlled to be equivalent for all TTSs to enable them to be fairly evaluated. The spectral features for the three TTSs were LSF with an order of 24. The HMM-based TTS also used the deltas of LSF. The excitation parameters for HMM-based TTS were composed of loga-

rithmic F0 and their corresponding delta coefficients. The frame lengths ware 20 ms and the update intervals ware 5 ms. The context-dependent HMM used three states for one phoneme, which was the same as the number of event targets for one phoneme that was used in the proposed HTD. Other parameters of the HMM-based TTS for Vietnamese were adopted from the original work by Vu et al. [13], while those of HTT were adopted from the original work by Qian et al.[4] and those of unit selection TTS were adopted from the original work in[14].

STRAIGHT version 4 [10] was used as a vocoder to generate the output waveforms. All parameters used for extracting F0, aperiodicity (AP), and spectral envelope with STRAIGHT were default parameters except for fs, frame size and frame step.

3.3. Subjective evaluations

For evaluating the TTS, subjective tests on intelligibility and naturalness were conducted. Five subjects who are native Vietnamese with normal hearing were required to attend the subjective tests.

Semantically unpredictable sentences (SUS) have been used as a standard measure to evaluate the intelligibility of a TTS, but there are no designs on Vietnamese SUS sentence lists at present. Therefore, a dataset of 20 testing sentences were chosen for the intelligibility evaluation with four restricted rules of preventing the subjects from predicting the meanings easily (rules 1–4), and two restricted rules for ensuring the reliability of the evaluation (rules 5–6):

(1) The Vietnamese words in the testing sentences were all low frequency;

(2) Only sentences composed from monosyllabic words were used to avoid ubjects from predicting the meaning of complex words with only their component words;

(3) Repeating the words between testing sentences is avoided, in order to prevent subjects remembering the words they heard previously;

(4) The sentences with less semantic relations were selected to avoid subjects predicting the meaning of the sentence;

(5) The sentences covering all Vietnamese tones and minimizing the repetition of tonal phonemes were selected;

(6) Only short sentences were selected to avoid the difficulty of subjects remembering the syllables that they heard in the testing sentence. The intelligibility scores were measured by word error rates (WER) of SUS sentences.

The naturalness of TTS has been widely evaluated by mean opinion scores (MOS). Therefore, MOS scores were used to evaluate the overall impression of naturalness of TTSs with a testing dataset that contained 20 long sentences with an average length approximately 25 syllables. Evaluations with long sentences were used to measure the speech naturalness in terms of both voice quality and segmental duration and timing.

The two testing datasets were chosen from the set of sentences that were not used for training, or concatenating, or rendering the TTSs.

The results of the intelligibility evaluations are shown in Table. 1. These results shows that the WERs of speech synthesized by HMM-based TTS, that of speech synthesized by the proposed HTD, and that of speech analyzed / synthesized by MRTD-STRAIGHT were very small. *The intelligibility of HMM-based TTS and HTD were equivalent and they highly outperformed the intelligibility of HTT and unit selection TTS.*

The results of the naturalness evaluation are shown in Fig. 4. *These results in terms of naturalness show that the proposed HTD was superior to all the HMM-based TTS, the unit selection TTS and the HTT.*

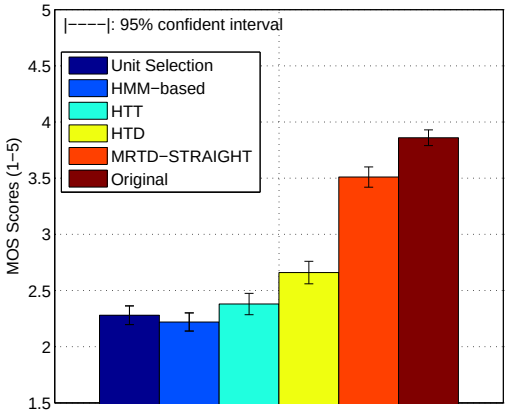


Figure 4: Naturalness mean scores and 95% confidence intervals for long sentences

It also reveals that the naturalness of the unit selection TTS was just slightly superior to that of the HMM-based TTS when evaluating with very long sentences due to its unstability on segmental duration and timing. The naturalness of the HTT also slightly outperformed the unit selection and HMM-based TTS. Therefore, although the naturalness of HTT is significantly high with huge amount of data for rendering [4], it is significantly reduced under limited data conditions.

The results from the intelligibility evaluation were consistent with the results from a HMM-based TTS [13] where the intelligibility scores of a Vietnamese TTS could reach 100%. The intelligibility of the mono-syllabic Vietnamese speech seems to be higher than that of other languages.

MOS score of 3.8 for the original speech was quite low since corpus DEMEN567 was not well recorded due to a low sampling frequency of 11025 Hz and the recording environment. The MOS score of speech analyzed / synthesized by MRTD-STRAIGHT was lower than that of the original speech due to the reconstruction errors of MRTD and STRAIGHT. The MOS scores for speech synthesized by all synthesizers were not compared with that of speech analyzed / synthesized by MRTD-STRAIGHT and the original speech, which are the two ideal limitations of HTD and unit selection TTS, HTT since they were implemented under a “limited data condition”.

3.4. Discussions on differences between the proposed HTD and HTT

Although the proposed HTD shares some common procedures with HTT [4], their concepts are completely different. These

Table 1: Word Error Rates (%)

	HMM	HTT	Unit Selection	HTD	MRTD-STRAIGHT	Original
Mean	0.25	3.82	10.83	0.25	0.25	0
95% confidence	0.09	0.71	0.88	0.09	0.09	0

differences are presented and discussed in this section. There are three main differences:

(1) HTT replaces all frames of the guided trajectory generated by HMMs with the closest frames found in the original database. Therefore, HTT can be considered to be one kind of unit selection that uses HMM-based TTS as an intermediate procedure to compute the target cost, resulting in improved stability in synthesized trajectory of speech. However, HTT shares several common disadvantages with unit selection TTS, e.g., their requirements for huge amounts of data for selection or rendering, their huge footprints, their high computational load, and their inflexibility for voice transformations.

The proposed HTD uses HMM-based TTS to generate spectral and prosodic trajectories. The spectral trajectory is then decomposed into its event functions and event targets. The prosodic trajectories and the event functions of the spectral trajectory are preserved to maintain the high intelligibility of HMM-based TTS, while the sparse event targets are replaced with the event targets located at the closest frames found in the original database to reduce over-smoothness in spectral sequence. Therefore, the proposed HTD is one extended version of HMM-based TTS in which speech synthesized by HMMs is transformed to the original speech by using MRTD, resulting in the improvement of synthesized speech in terms of naturalness while preserving main advantages of HMM-based TTS.

(2) HTT requires a huge database for rendering to ensure the smoothness of the synthesized speech because limited data may cause mismatches and discontinuities between consecutive frames. The smoothness of the synthesized trajectory in the proposed HTD is ensured by the smoothness of event functions and the stability and smoothness of the trajectory generated by HMM-based TTS. Therefore, the matching level of the "target selection" task does not strictly require precision as in HTT. As a result, the proposed HTD can synthesize stable and smooth speech even under limited data conditions.

(3) HTT can be combined with voice transformation by using multiple huge target databases for rendering. The requirement for huge target databases is not convenient for practical voice transformations where only a few target data are available. TD-based voice transformations [7], [8] could efficiently transform speaker individuality by preserving the event functions of source speech and transforming its event targets to those of target speech. This manner is similar to the proposed HTD, when event functions of spectral sequence synthesized by HMM-based TTS are preserved and its event targets are selected from an original database, or are transformed to those of the original speech. Therefore, it is possible to develop the proposed HTD to synthesize multiple voices with a multiple-voices database. The experimental results in this paper revealed that the proposed HTD was efficient with a small database. Therefore, the proposed HTD can be developed for voice transformations with limited target data. Although the proposed HTD was just evaluated with a single-voice database, it will be implemented with multiple-speakers and multiple-styles databases in the future to confirm its flexibility for voice transformations.

4. Conclusions

In this paper, a hybrid TTS among unit selection, HMM-based TTS, and MRTD, named HTD, was proposed. The experimental results show that the proposed HTD could borrow both the high intelligibility of HMM-based TTS and the high naturalness of unit selection TTS under limited data conditions. In the future, we will investigate other possible advantages of the proposed HTD such as the flexibility for voice transformations. We will also implement the proposed method with other languages to confirm the unification and language-independence of the proposed TTS.

5. Acknowledgements

This study was supported by the Grant-in-Aid for Scientific Research (A) (No. 25240026) and the A3 Foresight Program made available by the Japan Society for the Promotion of Science (JSPS).

6. References

[1] A.J. Hunt, A.W. Black, "Unit selection in a concatenative speech synthesis system using a large speech database," *Proc. ICASSP 96,* **1**, pp. 373–376, (1996).

[2] H. Zen, T. Toda "An overview of Nitech HMM-based speech synthesis system for Blizzard Challenge 2005," *Proc. Interspeech,* (2005).

[3] T. Toda, K. Tokuda, "A Speech Parameter Generation Algorithm Considering Global Variance for HMM-Based Speech Synthesis," *IEICE Trans. Inf. and Syst.,* Vol. E90-D, Issue 5, pp. 816-824, (2007).

[4] Y. Qian, F. K. Soong, Z. Yan, "A Unified Trajectory Tiling Approach to High Quality Speech Rendering," *IEEE Trans. on Audio, Speech, and Language Proc.,* Vol. 21, No. 2, pp. 280–290, (2013).

[5] R.B. Chicote, J. Yamagishi, S. King, J.M. Montero, J.M. Guarasa, "Analysis of statistical parametric and unit selection speech synthesis systems applied to emotional speech," *Speech communication,* Vol. 52, pp. 394404, (2010).

[6] PC. Nguyen, T. Ochi and M. Akagi, "Modified restricted temporal decomposition and its application to low rate speech coding," *IEICE Trans. Inf. and Syst.,* E86-D3 (2003).

[7] P.N. Binh and M. Akagi, "Efficient modeling of temporal structure of speech for applications in voice transformation," *Interspeech 2009,* pp. 1631-1634, (2009).

[8] V. Popa, J. Nurminen, M. Gabbouj, "A Novel Technique for Voice Conversion Based on Style and Content Decomposition with Bilinear Models," *Interspeech 2009*, pp. 2655-2658, (2009).

[9] B.S. Atal, "Efficient coding of LPC parameters by temporal decomposition," *Proc. ICASSP-83,* pp. 81–84 (1983).

[10] H. Kawahara, "STRAIGHT, Exploration of the other aspect of VOCODER: Perceptually isomorphic decomposition of speech sounds," *Acoust. Sci & Tech.,* **27**(6), 349–353 (2006).

[11] H. Phe, *Chinh ta Tieng Viet (Vietnamese Grammar),* (Da Nang Publisher), pp. 9–15, (2003).

[12] L.C. Mai and D.N. Duc, "Design of Vietnamese speech corpus and current status," *Proc. ISCSLP-06*, pp. 748–758 (2006).

[13] TT. Vu, MC. Luong and S. Nakamura, "An HMM-based Vietnamese speech synthesis system, Speech Database and Assessments," *Proc. COCOSDA-2009,* pp. 116–121 (2009).

[14] T.V. Do, D.D. Tran, and T.T. Nguyen, "Non-uniform unit selection in Vietnamese speech synthesis," *Proc. SoICT '11,* pp. 165-171, (2011).

Wavelets for intonation modeling in HMM speech synthesis

Antti Suni[1]*, Daniel Aalto*[1]*, Tuomo Raitio*[2]*, Paavo Alku*[2]*, and Martti Vainio* [1]

Institute of Behavioural Sciences (SigMe Group), University of Helsinki, Finland
[2]Department of Signal Processing and Acoustics, Aalto University, Finland
antti.suni@helsinki.fi, daniel.aalto@helsinki.fi, tuomo.raitio@aalto.fi
paavo.alku@aalto.fi, martti.vainio@helsinki.fi

Abstract

The pitch contour in speech contains information about different linguistic units at several distinct temporal scales. At the finest level, the microprosodic cues are purely segmental in nature, whereas in the coarser time scales, lexical tones, word accents, and phrase accents appear with both linguistic and paralinguistic functions. Consequently, the pitch movements happen on different temporal scales: the segmental perturbations are faster than typical pitch accents and so forth. In HMM-based speech synthesis paradigm, slower intonation patterns are not easy to model. The statistical procedure of decision tree clustering highlights instances that are more common, resulting in good reproduction of microprosody and declination, but with less variation on word and phrase level compared to human speech. Here we present a system that uses wavelets to decompose the pitch contour into five temporal scales ranging from microprosody to the utterance level. Each component is then individually trained within HMM framework and used in a superpositional manner at the synthesis stage. The resulting system is compared to a baseline where only one decision tree is trained to generate the pitch contour.

Index Terms: HMM-based synthesis, intonation modeling, wavelet decomposition

1. Introduction

The fundamental frequency (f_0) contour of speech contains information about different linguistic units at several distinct temporal scales. Likewise prosody in general, f_0 is inherently hierarchical in nature. The hierarchy can be viewed in phonetic terms as ranging from segmental perturbation (i.e., microprosody) to a levels that signal phrasal structure and beyond (e.g., utterance level downtrends). In between there are levels that signal relations between syllables and words (e.g., tones and pitch accents). Consequently, the pitch movements happen on different temporal scales: the segmental perturbations are faster than typical pitch accents, which are faster than phrasal movements and so on. These temporal scales range between several magnitudes from a few milliseconds to several seconds and beyond.

In HMM-based speech synthesis paradigm, all modeling is based on phone sized units. In principle, slower intonation patterns are more difficult to model than segmentally determined ones. Moreover, the statistical procedure of decision tree clustering highlights instances that are more common, resulting in a good reproduction of microprosody and overall trends (such as general downtrends) and relatively poor reproduction of prosody at the level of words and phrases. This shortcoming calls for methods that take into account the inherent hierarchical nature of prosody.

Traditionally the problem has been approached by using superpositional models which separate syllable and word level accents from phrases [2, 7]. On feature extraction side, discrete cosine transform parameterization of f_0 has been investigated, providing compact representation of the pitch contour [12]. Typically, each voiced segment or syllable and phrase are parameterized with a constant number of DCT coefficients, statistical clustering is performed based on contextual features, and synthesis is performed in additive fashion [11]. However, the constant number of coefficients is problematic for variable length units, and natural continuity between units is difficult to achieve.

In HMM framework, decomposition of f_0 to its hierarchical components during acoustic modeling has been investigated [4, 15]. These approaches rely on exposing the training data to a level-dependent subset of questions for separating the layers of the prosody hierarchy. The layers can then be modeled separately as individual streams [4], or jointly with adaptive training methods [15]. Results indicate that syllable level modeling improves prosody whereas higher levels do not provide benefits.

In HMM-based speech synthesis, f_0 is modeled jointly with voicing decision. The unit of modeling is typically a phone HMM with five states. For each state, predefined contextual questions concerning phones, syllables, words and phrases are used to form a set of possible splits in a decision tree. The splitting decisions are made in a greedy fashion based on likelihood increase. Thus the hierarchical nature of intonation is only implicitly addressed by questions on different levels of hierarchy. With multiple levels, including voicing decision, modeled by a single set of trees, the rare or slow events can not be modeled robustly, due to fragmentation of the training data by previous, more urgent splits for the short time scale of the model.

In this paper, we present a solution to the problems outlined above based on continuous wavelet transform (CWT). The CWT is used to decompose the f_0 contour into several temporal scales that can be used to model the levels ranging from microprosody to the utterance level separately. As well as separating the contour into meaningful temporally assigned levels – ranging from microprosody to utterance level prosody – the CWT produces a continous f_0 contour which has further merits. Earlier, wavelets have been used in speech synthesis context for parameter estimation [3, 6, 10].

We chose four f_0 modeling methods for comparison: (1) The normal HTS method using the MSD stream, and two

wavelet-based setups modeling the f_0 contour on several distinct levels: (2) one with a joint model and (3) one where five separate CWT based levels are modeled separately. In addition, (4) a continous interpolated f_0 stream model was added. The fourth method was added in order to evaluate the wavelet based methods against another model using continuous trajectories since interpolation alone has been reported to improve f_0 modeling [14].

Objective comparison of the proposed methods is presented against single-stream baselines using two GlottHMM [9] Finnish voices trained from a male and a female corpus.

2. Pitch decomposition and wavelets

2.1. Extraction and preprocessing of f_0

GlottHMM vocoder was used for estimating the fundamental frequency (f_0) of speech. GlottHMM is a physiologically oriented vocoder that uses glottal inverse filtering for separating speech into the glottal source signal and the vocal tract filter. The iterative adaptive inverse filtering (IAIF) method is used for the separation, and the f_0 is estimated from the glottal source signal that is free from the distracting vocal tract resonances [9].

The autocorrelation method [8] was used to estimate the f_0. A range of possible f_0 values is defined based on the speaker's f_0 range in order to reduce gross errors. The voiced-unvoiced decision is made based on the energy of the low frequency band (0–1 kHz) and the number of zero-crossings in the frame. The length of the frame from which the f_0 is estimated is longer than the speech analysis frame in order to estimate the lowest possible f_0 values, as low as 30 Hz. The frames determined as unvoiced are marked as zeros. Parabolic interpolation was used in order to reduce the estimation error due to finite sampling period; a quadratic function is fitted to the peak of the autocorrelation function (ACF) to find the refined f_0 value.

Finally, post-processing is applied to the estimated f_0 trajectory. A repetitive process is applied which consists of 3-point median filtering, filling small unvoiced gaps and removing outlier voiced sections, and detection of unnatural discontinuities based on weighted linear estimation of each individual f_0 estimate from previous and following samples. If the difference between the estimated and the actual values is greater than a specific threshold (based on the mean and variance of the f_0 trajectory), the original value may be replaced with a secondary f_0 estimate from the ACF. This replacement depends on the goodness of the fitting and the relative jump of the original f_0 estimate. An example of extracted f_0 is shown in the top pane of Figure 1.

2.2. Completion of f_0 over unvoiced passages

The wavelet method is sensitive to the gaps in the f_0 contour and therefore, the f_0 contour is completed to yield a continuous f_0 trajectory. Since the wavelet approach aims at connecting the signal to the perceptually relevant information, the linear frequency scale is transformed to the logarithmic semitone scale. A simple linear interpolation method is used. First, smoothed version of the original f_0 was created, and then interpolated over unvoiced passages. The smoothed unvoiced parts are then added to the original f_0 with 3 point median smoothing to reduce discontinuities in voicing boundaries. In addition, to alleviate edge artifacts, constant f_0 was added prior to and after the utterance. The pre-utterance f_0 value was set to the

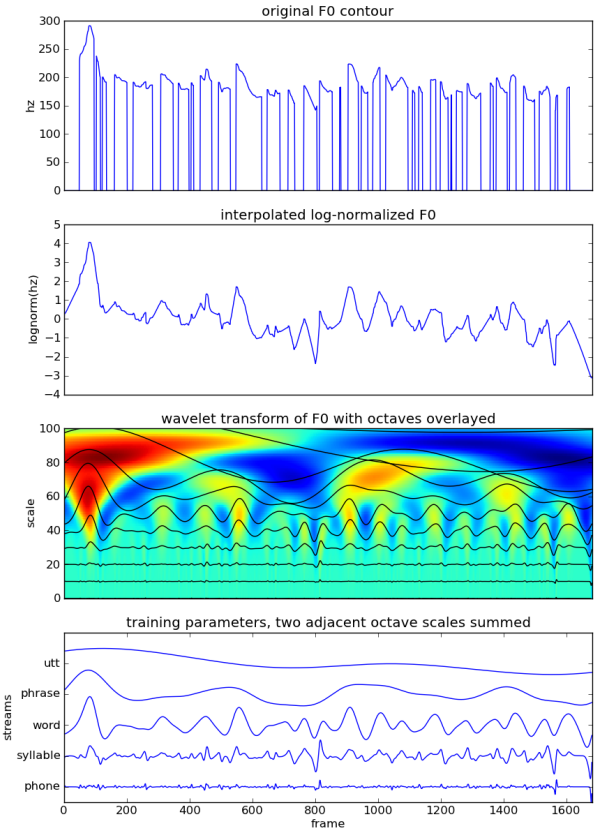


Figure 1: Example of f_0 parameterization. Top pane depicts the baseline method, *base*, in linear frequency scale; the second pane shows the interpolated baseline, *contf0*; third pane shows the continuous wavelet transform of the f_0 signal with the ten chosen scales separated by an octave (method *wave1*); the bottom pane shows the five scales that are merged from the continuous wavelet picture forming the basis of *wave5*

mean f_0 value calculated over the first half (in seconds) of the utterance; the post-utterance f_0 was set to the respective minimum. Finally, the interpolated $\log f_0$ contour is normalized to zero mean, unit variance as required by wavelet analysis. An example of an interpolated pitch contour is depicted in the second pane of Figure 1.

2.3. Wavelet based decomposition of f_0 contour

Wavelet transforms can be used to decompose a signal into frequency components similar to the Fourier transform. Although several alternatives exist, here we have chosen to use continuous wavelet transforms for f_0 decomposition. To define the wavelet transform, consider a (bounded) pitch contour f_0. The continuous wavelet transform $W(f_0)(\tau, t)$ of f_0 is defined by

$$W(f_0)(\tau, t) = \tau^{-1/2} \int_{-\infty}^{\infty} f_0(x) \psi \left(\frac{x - t}{\tau} \right) dx$$

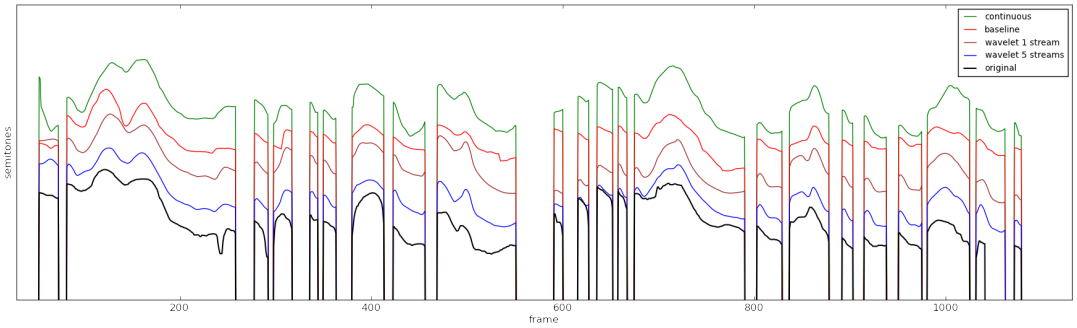


Figure 2: Example of synthesized f_0 contours with evaluated methods on a female corpus test utterance, overlaid three semitones apart.

where ψ is the Mexican hat mother wavelet. The original signal f_0 can be recovered from the wavelet representation $W(f_0)$ by inverse transform (for the proof, see [1, 5]):

$$f_0(t) = \int_{-\infty}^{\infty} \int_{0}^{\infty} W(f_0)(\tau, x)\tau^{-5/2}\psi\left(\frac{t-x}{\tau}\right) dx d\tau.$$

However, the reconstruction is incomplete, if all information on $W(f_0)$ is not availabe. Here, the decomposition and reconstruction is approximated by choosing ten scales, one octave apart. f_0 is represented by the wavelets as ten separate streams given by

$$W_i(f_0)(t) = W(f_0)(2^{i+1}\tau_0, t)(i+2.5)^{-5/2} \quad (1)$$

where $i = 1, \ldots, 10$ and $\tau_0 = 5$ ms, and the original signal is approximately recovered by

$$f_0(t) = \sum_{i=1}^{10} W_i(f_0)(t) \; + \; \epsilon(t) \quad (2)$$

where $\epsilon(t)$ is the reconstruction error. The reconstruction formula (2) is *ad hoc* and no attempts were made in this stage to optimize the computational efficiency. The accuracy of the reconstruction was evaluated by decomposing and reconstructing ten utterances spoken by a male and a female. The correlation between the original and the reconstructed f_0 signal was $99.7\,\%$ with root mean square reconstruction error of $1.03\ Hz$.

The continuous wavelet transform and ten distinct scales are shown in the third pane of the Figure 1. The scales 0 and 1 correspond to phone level (50 and 25 Hz), scales 2 and 3 correspond to syllable level (6 and 13 Hz), scales 4 and 5 show word level (1.6–3 Hz), scales 6 and 7 correspond to phrase level (0.4–0.8 Hz), and scales 8 and 9 correspond to utterance level. The adjacent scales are combined and shown in the bottom pane of the Figure 1. These five broad scales are separated by two octaves from each other. The correspondance of the prosodic levels of hierarchy and the wavelet scales is approximative and the wavelet scales are not adjusted to optimize the fit. Hence, e.g., not all the syllables have a duration that would fall in the "syllable scale".

3. Constructing the synthesis

3.1. Speech material

In order to carry out evaluation of the proposed f_0 modeling methods, two Finnish HMM-voices were trained, a male and a female one. The male database (MV) used is a traditional synthesis corpus, with rather carefully articulated set of 692 isolated sentences, while the female one (HK) is more diverse, consisting of 600 phonetically rich sentences as well as continuous prosodically rich read speech; 266 long sentences of fact and 607 sentences of diverse prose. 92 sentences of the male database was left out for evaluation purposes and 60 utterances of prose for the female. Both corpora have been tagged for word prominence on discrete scale ranging from 0 to 3, using acoustic features [13]. The prominence labels were used in both training and evaluation as contextual features. Thus the evaluation was not affected by TTS symbolic prosody prediction errors. In addition to word prominence, full context labels were generated with conventional features: quinphones with positional and length features of phones, syllables, word and phrases. Notably, more enriched labeling above word level would have been preferable for the current topic of modeling the prosodic hierarchy.

3.2. Parameterization of f_0 contours

Four different HMM-based statistical models for f_0 generation were compared. Synthesized f_0 contours based on these four and the original sentence f_0 are depicted in Figure 2.

3.2.1. base

A standard MSD model for f_0 is trained where each continuous f_0 passage between unvoiced segments is independently generated.

3.2.2. wave5

In the model *wave5*, five different f_0 components $w_1, \ldots, w_5$, defined by

$$w_i(t) = W_{2i-1}(f_0)(t) + W_{2i}(f_0)(t),$$

are independently trained by HMMs.

3.2.3. wave1

The different time scales correlate especially with their neighbors, so a plausible alternative would be to jointly model all the scales. This is done in *wave1* where one vector $V(t) = \{W_i(f_0)(t)\}_{i=1}^{10}$ contains the time scales.

3.2.4. *contf0*

Since the wavelet based methods *wave5* and *wave1* generate a continuous f_0 trajectory, and since interpolating the pauses in the training data improves the synthesized contours [14], an alternative, *contf0*, is offered where the unvoiced segments are interpolated in the same way as in the preprocessing of the wavelets.

3.3. HMM-training

The speech was parameterized with GlottHMM vocoder [9], yielding a 5-stream HMM structure: vocal tract spectrum LSFs and Gain (31 parameters), voice source spectrum LSFs (10), Harmonic-to-noise ratio (5) and $\log f_0$ (1). f_0 was then processed as described in the previous chapter. 5 streams (1 parameter each) for method wave5, 1 stream (10) for wave1 and one stream for continuous $\log f_0$. The baseline f_0 method was modeled as an MSD stream, others as continuous streams. With dynamic features further added, HMM training was perfomed in a standard fashion using HTS [16]. Stream weights affecting model alignment were set to zero for all streams except vocal tract spectrum LSFs and $\log f_0$. Decision tree clustering was perfomed individually for each stream without stream-dependent contextual question sets. Using the MDL criterion on decision tree building, the wave5 trees tended to become very large compared to baseline. Attempts were made to control the tree size with minimum leaf occupancy count, which was set to 10 on baseline MSD $\log f_0$ stream and 20, 25, 30, 60 and 70 for respective wave5 streams. In addition, MDL factor was set to 0.6 for $\log f_0$ stream and 1.5 for wave5 streams.

4. Evaluation

4.1. Evaluation data

The fundamental frequency parameters of the test utterances were generated from HMMs using original time alignments. For wavelet methods, the f_0 trajectories were constructed from generated scales using Equation (1). Voicing decision for continuous f_0 methods was based on the base MSD stream as well as mean and variance of f_0 for normalized wavelet methods.

The alignments were aquired by force-alignment method with the monophone models estimated during synthesis training. The synthesized sentences were checked manually for gross timing errors, and bad ones were excluded. The final MV test data consisted of 41 isolated utterances, spoken in the same formal style as the training data. By contrast, the HK test utterances consisted of 60 sentences of expressive prose.

4.2. Performance measures

The synthesized f_0 contours were compared to the original f_0 contours, estimated with GlottHMM, by measuring the correlation between the two curves and by calculating the root mean square error for each test utterance. Within an utterance, only the frames that were voiced with all methods were included. Also, due to frequent creaky voice with erratic pitch on original trajectories, the frames where the distance between original and at least one of the synthesized trajectories was more than 8 semitones, were excluded as outliers. It should be noted that these frames were completely excluded from the evaluation so that the comparisons were performed on exactly the same data sets. For the error calculation, the f_0 was converted to semitone scale with base 40 Hz. A Wilcoxon signed rank test was used to assess the statistical significance of the results.

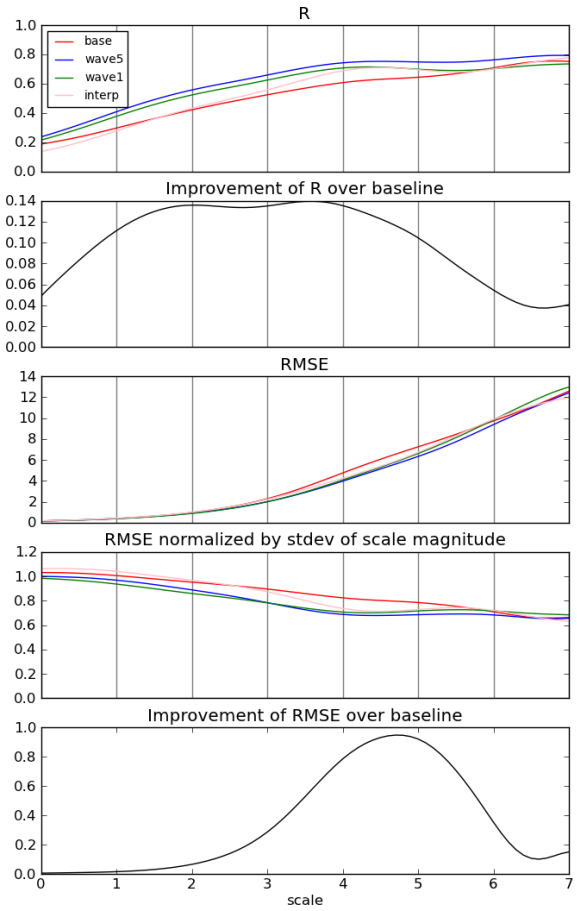


Figure 3: Evaluation results shown scale by scale. The top pane shows the correlations between the four synthesized contours and the original; second pane depicts the difference between the wavelet method *wave5* and *base*; third pane shows the absolute RMSE; in the fourth pane, the values are normalized by the variation at the scale; the bottom pane shows the difference in RMSE between the *wave5* and *base*.

4.3. Performance results

The correlations between the generated f_0 values and original contours showed significantly better performance for wavelet methods than for the baseline for both speakers. For the female data, the correlations over the test utterances were 0.76, 0.72, 0.72, and 0.68 for *wave5, wave1, contf0*, and *base*, respectively, as shown in Table 1. The *wave5* was better than *wave1* ($V = 1298$, $p < 0.05$), better than *contf0* ($V = 1324$, $p < 0.05$) and *base* ($V = 1445$, $p < 0.005$). In addition, the *wave1* was better than *base* ($V = 1329$, $p < 0.05$) but not significantly different

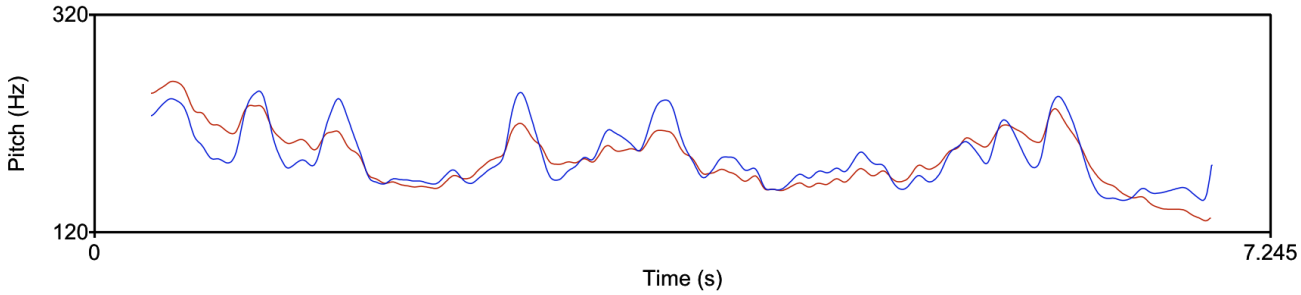


Figure 4: The reconstruction can be weighted to enhance the word level (blue curve) or the phrase level (red curve) intonation.

from *contf0* ($V = 1064$, $p > 0.1$). The *contf0* was marginally better than the *base* ($V = 702$, $p < 0.1$).

The male data showed similar patterns. The correlations over the test utterances were 0.85, 0.84, 0.81, and 0.81, respectively. The *wave5* was marginally better than *wave1* ($V = 288$, $p < 0.1$), better than *contf0* ($V = 129$, $p < 0.001$) and *base* ($V = 88$, $p < 0.001$). In addition, the *wave1* was better than *base* ($V = 136$, $p < 0.001$) and *contf0* ($V = 196$, $p < 0.005$). The *contf0* and the *base* were not significantly different ($V = 439$, $p > 0.1$).

Table 1: *A summary of the performance results of the syntheses. The means of the performance measures for each of the two data sets (female, male).*

	wave5	wave1	contf0	base
corr (F)	0.76	0.72	0.72	0.68
corr (M)	0.85	0.84	0.81	0.81
RMSE (F)	1.38	1.44	1.48	1.53
RMSE (M)	1.57	1.60	1.75	1.76

The root mean square error patterns are similar to the correlation results of the previous paragraphs. For the female data, the root mean square errors were 1.38, 1.44, 1.48, and 1.53 semitones for *wave5, wave1, contf0* and *base*, respectively. The *wave5* outperformed the *wave1* ($V = 1551$, $p < 0.001$), the *contf0* ($V = 1666$, $p < 0.001$, and the *base* ($V = 1781$, $p < 0.001$). The *wave1* and the *contf0* were statistically not different ($V = 1085$, $p > 0.1$), but the *wave1* was better than the *base* ($V = 1419$, $p < 0.005$). The *contf0* was better than *base* ($V = 599$, $p < 0.01$). For the male data, the root mean square error was 1.57, 1.60, 1.75, and 1.76 semitones for *wave5, wave1, contf0* and *base*, respectively. The *wave5* was not different from the *wave1* ($V = 307$, $p > 0.1$) but was better than the *contf0* ($V = 143$, $p < 0.001$) and the *base* ($V = 96$, $p < 0.001$). The *wave1* outperformed both the *contf0* ($V = 206$, $p < 0.005$) and the *base* ($V = 145$, $p < 0.001$). Finally, the *contf0* and *base* did not differ significantly ($V = 433$, $p > 0.1$).

4.4. Temporal scale analysis of the results

In Figure 3, the performance measures over the female test sentences are decomposed to the scale-wise components. Overall, the *wave5* is better than the baselines at all scales. However, the difference is pronounced for the middle scales.

5. Discussion and conclusions

The results of the objective evaluation are in line with previous research. Continuous f_0 modeling is found significantly better than the standard HTS method. On male voice, the synthesis of f_0 is very accurate, suggesting that existing methods are capable of modeling higher level structures to an adequate degree, given consistent style and accurate labels of word prominence. Consequently, the differences between evaluated methods are rather small, though the wavelet based methods provide some gains. As expected, the performance of all evaluated methods is lower on female voice due to difficult test utterances of continuous expressive prose, and also possibly due to more errors in f_0 estimation during analysis. Here, the individually modeled wavelet scales provide a large improvement. However, subjective evaluation is still required for final conclusions.

Overall, the results suggest that the proposed method largely solves the fragmentation problem caused by simultaneous decision tree clustering of all levels of prosodic hierarchy. Yet, somewhat contrary to expectations the improvements seem larger on word level and syllable level than on phrase level. Although technical problems of higher scales affected by boundary effects on wavelet analysis may have an effect, this mainly highlights the need for new contextual features on supra-word level, beyond position and number. With the proposed method the features representing for instance constituent structure, phrase type and utterance modality could actually have an effect on the synthesized prosody.

The wavelet decomposition offers a possibility of adjusting the weights of individual scales prior to reconstruction. This could have potential applications in speaking style modification. For example, informal listening suggested that increasing the weight of the word level makes the synthesized speech sound more resolute and perhaps more intelligible, while listening longer passages is less displeasing when phrase level is emphasized. Moreover, moderate modifications do not seem to have adverse effect on naturalness. Figure 4 presents an example of this type of modification. Local weighting within utterance could also be applied for e.g. emphasis reproduction. Rapid adaptation of speaking style based on transform of the scale weights alone could also be considered.

The current paper has presented a novel method of f_0 modeling based on wavelet decomposition. Many open questions remain. Selection of scales and model structure were made based on intuition alone, no other wavelets beyond mexican hat were considered, neither more popular discrete wavelet transform.

Also, while the proposed method seems quite suitable for the current HMM-synthesis framework, it is deeply unsatisfying to model utterance level f_0 contour with inherently sub-segmental models, when the discrete cosine transform or discrete wavelet transform could represent the level with only a few coefficients.

6. Acknowledgments

The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7/2007–2013) under grant agreement n^{o} 287678 and the Academy of Finland grants 128204 and 125940.

7. References

[1] Daubechies, I., "Ten lectures on wavelets", Philadelphia, SIAM, 1992.

[2] Fujisaki, H., Hirose, K., Halle, P., and Lei, H., "A generative model for the prosody of connected speech in Japanese", Ann. Rep. Eng. Research Institute 30: 75–80, 1971.

[3] Kruschke, H. and Lenz, M., "Estimation of the parameters of the quantitative intonation model with continuous wavelet analysis", in Proc. Eurospeech'03, 4, pp. 2881-2884, Geneva, 2003.

[4] Lei, M., Wu, Y. J., Ling, Z. H., and Dai, L. R., "Investigation of prosodic F_0 layers in hierarchical F_0 modeling for HMM-based speech synthesis", Proc. IEEE Int. Conf. Signal Processing (ICSP) 2010, 613–616.

[5] Mallat, S., "A wavelet tour of signal processing", Academic Press, San Diego, 1998.

[6] Mishra, T., van Santen, J., and Klabbers, E., "Decomposition of pitch curves in the general superpositional intonation model", Speech Prosody, Dresden, Germany, 2006.

[7] Öhman, S., "Word and sentence intonation: a quantitative model", STLQ progress status report, 2–3:20–54, 1967.

[8] L. Rabiner, "On the use of autocorrelation analysis for pitch detection", IEEE Trans. Acoustics, Speech, and Signal Processing, vol. 25, no. 1, pp. 24–33, 1977.

[9] Raitio, T., Suni, A., Yamagishi, J., Pulakka, H., Nurminen, J., Vainio, M., and Alku, P., "HMM-based speech synthesis utilizing glottal inverse filtering", IEEE Trans. on Audio, Speech, and Lang. Proc., 19(1):153–165, 2011.

[10] van Santen, J. P. H., Mishra, T., and Klabbers, E., "Estimating phrase curves in the general superpositional intonation model", Proc. 5th ISCA speech synthesis workshop, Pittsburgh, 2004.

[11] Stan, A. and Giurgiu, M., "A Superpositional Model Applied to F0 Parameterization using DCT for Text-to-Speech Synthesis", Proceedings of 6th Conference on Speech Technologoy and Human-Computer Dialogue (SpeD), 2011, 1–6, Brasov.

[12] Teutenberg, J., Watson, C. I., and Riddle, P., "Modelling and synthesising F0 contour with the discrete cosine transform", ICASSP 2008, 3973–3976, 2008.

[13] Vainio, M., Suni, A., and Sirjola, P., "Accent and prominence in Finnish speech synthesis", Proc. 10th Int. Conf. Speech and Computer (Specom 2005), 309–312.

[14] Yu, K. and Young, S., "Continuous F0 Modeling for HMM based statistical parpametric speech synthesis", Trans. Audio, Speech and Lang. Proc., 19:5, 1071–1079, 2011.

[15] Zen, H. and Braunschweiler, N., "Context-dependent additive log F0 model for HMM-based speech synthesis", Proc. Interspeech 2009: 2091–2094.

[16] Zen, H., Nose, T., Yamagishi, J., Sako, S., Masuko, T., Black, A., Tokuda, K., 2007. The HMM-based speech synthesis system (HTS) version 2.0. In: SSW6. pp. 294–299.

A Common Attribute based Unified HTS framework for Speech Synthesis in Indian Languages

B. Ramani[1], S.Lilly Christina [1], G Anushiya Rachel [1], V Sherlin Solomi[1],
Mahesh Kumar Nandwana[2], Anusha Prakash[2], Aswin Shanmugam S [2], Raghava Krishnan[2]
S P Kishore[3], K Samudravijaya[4]
P Vijayalakshmi[1], T Nagarajan [1] and Hema A Murthy[2]

[1]Speech Lab, SSN College of Engineering, Chennai, India
[2] Department of Computer Science and Engineering, IIT Madras, India
[3] Speech and Vision Lab, IIIT Hyderabad, India
[4] TCS, TIFR Bombay, India Email: hema@cse.iitm.ac.in

Abstract

State-of-the art approaches to speech synthesis are unit selection based concatenative speech synthesis (USS) and hidden Markov model based Text to speech synthesis (HTS). The former is based on waveform concatenation of subword units, while the latter is based on generation of an optimal parameter sequence from subword HMMs. The quality of an HMM based synthesiser in the HTS framework, crucially depends on an accurate description of the phoneset, and accurate description of the question set for clustering of the phones. Given the number of Indian languages, building a HTS system for every language is time consuming. Exploiting the properties of Indian languages, a uniform HMM framework for building speech synthesisers is proposed. Apart from the speech and text data used, the tasks involved in building a synthesis system can be made language-independent. A language-independent common phone set is first derived. Similar articulatory descriptions also hold for sounds that are similar. The common phoneset and common question set are used to build HTS based systems for six Indian languages, namely, Hindi, Marathi, Bengali, Tamil, Telugu and Malayalam. Mean opinion score (MOS) is used to evaluate the system. An average MOS of 3.0 for naturalness and 3.4 for intelligibility is obtained for all languages.

1. Introduction

A successful technique for speech synthesis is the unit selection-based concatenative synthesis (USS). This system selects and concatenates pre-recorded speech units in the database such that the target and concatenation costs are minimised [1]. In order to obtain high-quality synthetic speech, the size of the database required is large, to ensure that sufficient examples for each unit in every possible context is available. This results in a large footprint for USS systems.

A recent approach to speech synthesis is statistical parametric synthesis. This method involves the generation of context-dependent HMMs which are concatenated to form a sentence HMM, corresponding to the input text provided. Unlike the USS approach, the prosodic characteristics of the voice can be modified by simply varying the HMM parameters [2],[3] thereby reducing the requirement for large amount of data requirement.

A HMM-based speech synthesis system requires the following: (a) text data in a language, (b) speech data corresponding to the text, (c) time-aligned phonetic transcription, (d) context-specific features for phones if they exist (e) a question set for tying phone models. (a) and (b) are language dependent, while the rest of the modules can be made language-independent in the Indian language context.

In India most languages can be classified as Aryan and Dravidian or a mix of both. In the current work six languages are chosen Hindi, Marathi, Bengali, Tamil, Malayalam and Telugu. Indian languages have several phonetic similarities among them [4], which suggests the possibility of a compact, common phone set for all the languages. Deriving time-aligned phonetic transcription is a tedious task. The acoustic similarity among the same phones of different languages leads to a set of common, context-independent set of acoustic models that can be used for segmenting the speech signal into phonemes. Further, a common question set can also be derived that can be used for clustering in the HTS framework. This is the primary motivation for this paper. This work is motivated by the efforts of [5] in the context of building synthesis for all the languages of the world. In this work, the focus is restricted to build systems for Indian languages which number about 1652 at the time of this writing [6]. The ultimate goal is to build a generic text-to-speech system for Indian languages which can be adapted for new languages using a small amount of adaptation data.

The paper is organised as follows: Section 2 describes the speech corpora. In order to provide the appropriate context, in Section 3 the HMM based speech synthesiser is described with particular emphasis on different modules. Section 4 discusses how a common phone set, common acoustic models, and a common question set are obtained. Section 5 describes the Indian language synthesiser. Section 6 gives the performance analysis and Section 7 concludes the paper.

2. Speech Corpora

In the work presented in [7], speech data is collected for six of the Indian languages, namely, Tamil, Malayalam, Telugu, Hindi, Marathi and Bengali for building an USS based system. 12 hrs of speech data is collected from a female speaker (voice talent) for each of the languages separately, in a studio environment at 16KHz, 16bits/sample. The data consists of sentences from short stories, novels, science, sports, and news bulletins.

3. HMM-Based Speech Synthesiser

HMM-based speech synthesis consists of a training and synthesis phase. In the training phase, spectral parameters, namely, Mel generalised cepstral coefficients (mgc) and their dynamic features, the excitation parameters, namely, the log fundamental frequency (lf0) and its dynamic features, are extracted from the speech data. Using these features and the time-aligned phonetic transcriptions, context-independent monophone HMMs are trained [2]. The basic subword unit considered for the HMM-based system is the context-dependent pentaphone. For Indian language synthesis too, the pentaphone is considered as the basic subword. The UTF-8 text is converted to a sequence of pentaphones. As in conventional HTS, the context-dependent models are initialised with a set of context-independent monophone HMMs. A sequence of steps based on the common question set, is used for state-tying, which results in tree based clustering of states[8].

In the synthesis phase, again as in conventional HTS, context-dependent label files are generated for the given text and the required context-dependent HMMs are concatenated to obtain the sentence HMM. Spectral and excitation parameters are generated for the sentence and a speech waveform is synthesised. This process is illustrated in Fig. 1 [2][1]. As mentioned in Section 1, except for the text and speech data, which are language-dependent, the rest of the modules can be made language-independent by preparing a common phone set and common question set. The following section describes the common attributes across different Indian languages that can be shared to prepare common phone set, acoustic models, and question set.

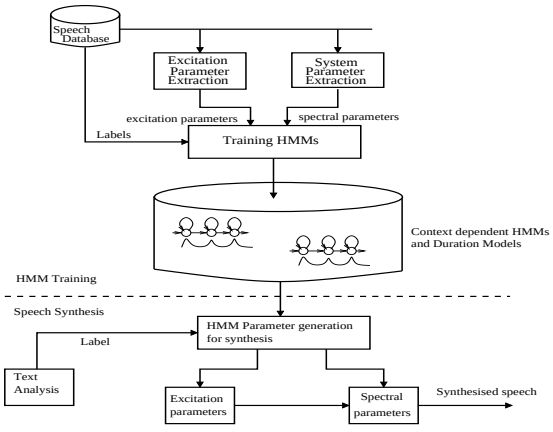


Figure 1: Overview of HMM-Based Speech Synthesis

[1]This figure has been redrawn from [2]

4. Sharing Common Attributes

4.1. Phone Set

A list of phones in each of the six languages considered in this paper, namely, Tamil, Malayalm, Telugu, Hindi, Marathi and Bengali, is shown in Figs. 2 and 3. A tag **v** is associated with some sounds for Tamil to denote the voiced counterparts. It is observed that there are 25 phones that are common to all six languages. Each language has 10 to 12 vowels, out of which 8 are common to all. Most consonants among different languages are observed to be phonetically similar. This is also confirmed by the studies of Prahalad et al. in [9]. 33 consonants are common to all languages, except Tamil, which has only 26 consonants as against 33 or 34 in others. For example, the short and long vowels for /a/, /i/ and /u/ are present in all the six languages. While Tamil, Malayalam and Telugu also have long vowels for /e/ and /o/, the Aryan languages have long vowels for /a/, /i/ and /u/ only. There are some sounds that are unique to some languages. The retroflex /zh/ is present only in Tamil and Malayalam. Tamil does not have any aspirated consonants. The palatals /c/, /ch/, /j/, /jh/ are all affricates in Indian languages. Most of the questions for Hindi were first derived from [10]. Given the common phoneset, it was observed that a Question Set of 53 questions is sufficient to cover the common phone set. Additionally, rules are included for specific languages. Altogether a set of 60 questions are prepared. The phones are grouped and mapped to labels that are closest based on the International Phonetic Alphabet (IPA). The IPA labels are modified to exclude special characters for convenience. The IPA labels map directly to the graphemes of each language (Figure 2 and 3) [2]. We now list the rationale for the development of the common phone set:

1. Rules for mapping similar sounds across languages: This common phoneset is a standard set of labels (in Roman script) for speech sounds commonly used in Indian languages. This document lists the label set for 13 languages (currently being processed by ASR/TTS consortia of TDIL, DIT, Government of India). These labels are used for computer processing of spoken Indian languages.
   (a) Similar sounds in different languages are given a single label.
   (b) The IPA symbol refers to an exemplar (Hindi/Tamil/other) language.
   (c) This is not an IPA chart of sounds of Indian languages.
   (d) The label set is designed such that the native script is largely recoverable from the transliteration.

   A label may consist of a sequence of alphanumeric characters of the Roman alphabet; they will not contain any special character such as quote, hyphen etc. All labels are in lower case even though

[2]This is a partial set. Mapping is available for all 13 languages from the authors

Label	IPA	Hindi	Marathi	Bengali		Tamil	Malayalam	Telugu
				P	G			
a	/a/	अ	अ	-	-	அ	അ	అ
ax	/ɔ/	-	ऑ	অ	অ	-	-	-
aa	/a: /	आ	आ	আ	আ	ஆ	ആ	ఆ
axx	/ɔ /	-	-	-	-	-	-	-
i	/ɪ /,/i/	इ	इ	ই,ি	ই	இ	ഇ	ఇ
ii	/i: /	ई	ई	-	ঈ	ீ	ഈ	ఈ
u	/u/,/ʊ /	उ	उ	উ,ু	উ	உ	ഉ	ఉ
eu	/ɯ /	-	-	-	-	உ	◌	-
uu	/u: /	ऊ	ऊ	-	ঊ	ூ	ഊ	ఊ
rq	-	ऋ,ृ	ऋ,ृ	ঋ,ৃ	ঋ,ৃ	-	ഋ	ఋ,ృ
e	/e/	-	-	এ	এ	எ	എ	ఎ
ee	/e: /	ए	ए,ऐ	-	-	ஏ	ഏ	ఏ
ei	/ɛ: /	ऐ	-	-	-	-	-	-
ai	/aI/	-	ऐ	-	-	ஐ	ഐ	ఐ
oi	/oj/	-	-	ঐ	ঐ	-	-	-
o	/o/	ओ	ओ,ओ	ও	ও	ஒ	ഒ	ఒ
oo	/o: /	-	-	-	-	ஓ	ഓ	ఓ
ae	/ae/	-	ऍ	-	অ্যা	-	-	-
au	/a ʊ /	-	औ	-	-	ஔ	ഔ	ఔ
ou	/oʊ /	औ	-	ঔ	ঔ	-	-	-
k	/k/	क	क	ক	ক	க	ക	క
kh	/kʰ /	ख	ख	খ	খ	-	ഖ	ఖ
g	/g/	ग	ग	গ	গ	ஸ்ர	ഗ	గ
gh	/gʱ /	घ	घ	ঘ	ঘ	-	ഘ	ఘ
ng	/ŋ /	ङ	ङ	ঙ,ং	ঙ	ங	ങ	ఙ
c	/ʧ/	च	च	চ	চ	ச	ച	చ
ch	/ʧʰ /	छ	छ	ছ	ছ	-	ഛ	ఛ
cx	/ʧ_̱ /	-	च	-	-	-	-	-
j	/dʒ/	ज	ज	জ,য	জ	ஜ	ജ	జ
jh	/dʒʱ /	झ	झ	ঝ	ঝ	-	ഝ	ఝ
jx	/dʒ_̱ /	-	ज	-	-	-	-	-
nj	/ɲ/	-	ञ	-	-	ஞ	ഞ	ఞ
tx	/ʈ/	ट	ट	ট	ট	ட	ട	ట
txh	/ʈʰ /	ठ	ठ	ঠ	ঠ	-	ഠ	ఠ
dx	/ɖ /	ड	ड	ড	ড	ட்ட	ഡ	డ
dxh	/ɖʱ /	ढ	ढ	ঢ	ঢ	-	ഢ	ఢ
nx	/ɳ /	ण	ण	ণ	ণ	ண	ണ	ణ
t	/t/	त	त	ত,ৎ	ত,ৎ	த	ത	త
th	/tʰ /	थ	थ	থ	থ	-	ഥ	థ
d	/d/	द	द	দ	দ	த்த	ദ	ద
dh	/dʱ /	ध	ध	ধ	ধ	-	ധ	ధ

Figure 2: Common Phone Set

Label	IPA	Hindi	Marathi	Bengali		Tamil	Malayalam	Telugu
				P	G			
n	/n/	न,ऩ	न,ऩ	ন	ন	ந,ன	ന	న
nd		-	-	-	-	ந்த	-	-
p	/p /	प	प	প	প	ப	പ	ప
ph	/pʰ /	फ	फ	ফ	ফ	-	ഫ	ఫ
b	/b/	ब	ब	ব	ব	ப்ப	ബ	బ
bh	/bʱ /	भ	भ	ভ	ভ	-	ഭ	భ
m	/m/	म	म	ম	ম	ம	മ	మ
y	/j/	य,य़	य,य़	য়	য	ய	യ	య
r	/r/	र,ऱ	र,ऱ	র	র	ர	ര	ర
l	/l/	ल	ल	ল	ল	ல	ല	ల
lx	/l_̱/	-	ळ,ळ	-	-	ள	ള	ళ
w	/ʋ /	व	व	-	ওয়	வ	വ	వ
sh	/ʃ /	श	श	শ,ষ	শ	-	ശ	శ
sx	/ʂ /	ष	ष	-	ষ	ஷ	ഷ	ష
s	/s /	स	स	স	স	ஸ	സ	స
h	/ɦ /	ह	ह	হ	হ	ஹ	ഹ	హ
kq	/q/	क़	क़	-	-	-	-	-
khq	/x/	ख़	ख़	-	-	-	-	-
gq	/ɣ /	ग़	ग़	-	-	-	-	-
z	/z /	ज़	ज़	ज़	-	-	-	-
jhq	/ʒ /	झ़	झ़	-	-	-	-	-
dxq	/ɽ/	ड़	ड़	ড়	ড়	-	ऽ	-
dxhq	/ɽʱ /	ढ़	ढ़	ঢ়	ঢ়	-	-	-
dhq		-	-	-	-	-	-	-
f	/f/	फ़	फ़	-	ফ়	◌ॅ◌	-	-
bq		-	-	-	-	-	-	-
yq		-	-	-	য়	-	-	-
nq		-	-	-	-	-	ന	-
rx	/ɾ /	-	-	-	-	ற	റ	ఱ
sq		-	-	-	-	-	-	-
zh	/ɻ /	-	-	-	-	ழ	ഴ	ళ
nxh	/ɳ_̱ʰ/	-	ण्ह	-	-	-	-	-
nh	/nʰ/	-	न्ह	-	-	-	-	-
mh	/mʰ/	-	म्ह	-	-	-	-	-
rh	/ɾʰ/	-	ऱ्ह	-	-	-	-	-
lh	/lʰ/	-	ल्ह	-	-	-	-	-
wh	/wʰ/	-	व्ह	-	-	-	-	-
q		◌ं	◌ं	-	◌ং	-	◌ം	◌ం
hq		◌ः	◌ः	-	◌ঃ	-	◌ഃ	◌ః
mq		◌ँ	◌ँ	◌ঁ	◌ঁ	-	-	-

Figure 3: Common Phone Set (contd.)

the labels are case insensitive. Since the number of speech sounds are larger than the Roman alphabet, a system of suffixes as well as letter combinations are used for labels.

2. Notes on suffixes:

   (a) Aspiration: Use suffix h to denote aspiration: k (क) versus kh (ख).

   (b) Retroflex consonants: Use suffix x to denote retroflex place of articulation: t (त) versus tx (ट).

   (c) Nukta/bindu: Use suffix q to denote a nukta/bindu: dx (ड) versus dxq (ड़). Nukta (a dot below the glyph) may denote a flap/tap or a fricative variant of the consonant. Bindu (a dot above a [vowel] glyph) denotes a nasal after the vowel; the place of articulation of the nasal will be the same as that of the following consonant. If there is no consonant after the bindu, the vowel is nasalized.

   (d) Nasalized vowel: Use suffix n to denote nasalization of a vowel: k a h aa (कहा) versus k a h aan (कहाँ).

   (e) Geminated sounds: The label for a geminated consonant is the label of the corresponding single consonant with the first letter of the label repeated. Example: p a k aa (पका) versus p a kk aa (पक्का) in Hindi; a dd aa (अद्दा) in Hindi; a ll a m (అల్లమ్) (ginger in Telugu).

   (f) Other special cases: Use suffix x to denote

certain special cases: reduced vowel (axx) in various languages; "a" of Bangla; apical affricates of Marathi; special r of Dravidian languages etc.

(g) Priority of suffixes: Some symbols may have multiple suffixes. In such cases the following is the priority (in decreasing order): x h q n

3. Notes on Matras, Diphthongs and Halant:

(a) The label for a vowel matra is the same as that of the vowel.

(b) The label of a diphthong is generated as a concatenation of the labels of the corresponding vowel. The exceptions to this rule are "ae", "ea" and "eu"; these are monophthongs.

(c) The halant in Indian scripts denotes the absence of the implicit "a" in Indian consonant characters. It is not a sound and hence there is no label for halant. The morphological analyser of the language deletes the implicit "a" when a halant is present in the script.

(d) Punctuation marks: The 'transliteration' module will retain the punctuation marks (exception: '|' and "||" will be replaced by fullstops); these are useful for prosody generation. The morphological analyser will remove the punctuation marks while generating the word/phone level transcription.

4. Language specific notes: North-eastern languages have sounds (and labels) specific to a subset of the languages. Wherever required, a set of additional phonemes are defined.

Ideally, once the phones are generalised, context-independent models of acoustically similar phones across languages, can be combined. A compact set of acoustic models can be obtained to derive the time aligned phonetic transcriptions for the speech data. As this is the first attempt at a common phoneset for Indian languages, individual phone models are built for each of the languages using the aligned data.

4.2. Question Set

A question set is the primary requirement for tree-based clustering in an HMM-based speech synthesis system. A decision tree, that is similar to a binary tree with a yes/no question at each node is used. Relevant linguistic and phonetic classifications are included which enable accurate acoustic realisation of a phone. The question at each node of the tree is chosen such that there is a gain in the likelihood. Depending upon the answers, the phonemes for every language are split into categories, which are then tied. In the common question set, 60 common questions are formulated. Given that a common phoneset has been defined, a common question set was prepared for the languages. This set is a super set of questions across all Indian languages. This common question set that has been tested for 13 Indian languages. The number of questions in the common question set is fixed regardless of the language, since irrelevant entries in the question set are ignored while clustering [8].

5. Indian Language Synthesiser

5.1. Data Preparation

The wave files and the corresponding label files are required for building the HTS system. The common phone set for all the six languages are derived as described in Section 4.1. Common acoustic models, five minutes of speech data (phonetically balanced) for the language Tamil is considered as a representative for Telugu and Malayalam. For Aryan languages Hindi is chosen as the starting point for Marathi and Bengali. To generate unique phoneme models for the rest of the languages, few sentences are chosen in each language. Time-aligned phonetic transcriptions are obtained for this data by segmenting manually at the phoneme level using visual representations such as waveforms and the corresponding spectrograms. Monophone HMMs for all the phonemes are generated using the label files obtained. With these models, forced Viterbi alignment is performed iteratively to segment the rest of the data. This work is distinctly different from polyglot synthesis as in [11, 12] in that no attempt is made to generate common phones across different languages using the monphones from multiple languages. It is an attempt like the global phone project as in [13] to quickly build speech recognisers for various languages. This is also unlike the effort in [14], where speakers are clustered to produce a monolingual synthesiser for a new language with little adaptation data.

The effort in this paper is to primarily address the non-availability of annotated data for Indian languages. Further, there are at best only small vocabulary isolated word/isolated phrase recognition systems. Therefore, to obtain good initial monophone HMMs, a small amount of data must be manually transcribed. To reduce the effort required in manual transcription, two languages Hindi (Aryan) and Tamil (Dravidian) are first chosen, for which about 5 mins of data is manually transcribed. For languages that have additional phones, a few sentences from the given language are transcribed. This data is used to initialise the monophone HMMs. These monophone HMMs are used to force-align all the data of the appropriate group the language belongs. The HMM models are iteratively re-estimated using embedded re-estimation. Ultimately a set of language dependent HMMs are produced.

Summarising:

1. Time-aligned phonetic transcriptions are derived, for 5 mins of Tamil/Hindi speech data (phonetically balanced) manually and few sentences from each language to include unique phonemes in each language, using visual represenations such as waveforms and the corresponding spectrograms.

2. Using this data, context-independent phonemes models are trained (isolated-style training)

3. Using these models and the phonetic transcriptions (using the common phone set), the entire speech data is segmented using forced-Viterbi alignment.

4. Using the newly derived time-aligned phonetic transcription (phone-level label files), new context-independent phoneme models are trained.

5. Steps 3 and 4 are repeated N times ($N = 5$, here).

6. After N iterations, the HMMs are used to segment the entire speech data, again. These boundaries are considered as final boundaries.

5.2. Experimental Setup

Developing an HMM-based speech synthesis systems involves a training and synthesis phase. The training phase primarily requires the utterance structures, that are derived from Festival[15]. A 105 dimensional feature vector consisting of Mel-generalised cepstral (mgc) coefficients (35) their delta (35), and acceleration coefficients (35), 3 dimensional excitation features, that is, log F0 and its dynamic features are extracted from the speech files. In addition to this, the utterances are used to obtain contextual features (53 features), namely, number and position of words, syllables and phonemes, phrase breaks, stress of the subword units, which are in turn used to generate the context-dependent label files. From the parameters extracted, four-stream monophone models with five states and a single mixture component per state are generated for all 39 phonemes in the database. Five-stream duration models, with a single state and a single mixture component per state, are also generated for each phoneme. Using the common question set, tree-based clustering is carried out and context-dependent models are trained.

6. Performance Evaluation

To give perspective to the HTS based system, the HMM based speech synthesis system is compared with the syllable-based USS systems developed in [7]. Since Indian languages are syllable-timed and syllable is the fundamental production unit [16, 17], it is shown in [7, 16, 18, 19] that syllable-based synthesisers can be built for Indian languages with a minimal question set in the festival [15] framework. The festival based speech synthesis systems can be accessed at `http://www.iitm.ac.in/donlab/festival/`. These systems have been developed by a consortium of 12 Institutions across India and represents a set of reasonable quality text-to-speech synthesis systems developed for Indian languages. The UTF-8 encoding of the graphemes that correspond to the syllables of a language are directly used to build the systems. Part-of-speech taggers, tones and break indices markers are not readily available for all the Indian languages. Therefore, they are not used in building the synthesis systems. Similar to cluster unit based concatenative synthesis, cluster units are made from a syllable inventory. A semiautomatic algorithm [20] is used to build an inventory of syllables. As festival is primarily phoneme-centric, festival had to be modified to use syllable as a fundamental unit. A set of hand-crafted rules were developed to cluster at the syllable-level. For example, the number of units in the leaf is reduced to 20, syllables are clustered based on their position in the word, optimal coupling is turned off, etc. For details, refer to the [7]. The number of frequently used syllables is no more than 300 [21], but the syllable distribution has long tails. Fallback units to the akshara (CV) and phonemes are provided. These units are obtained by force-alignment at the syllable-level.

6.1. Building HTS based systems

Initially HMM-based speech synthesis systems were built for only Tamil and Hindi using the common phone set and common question set. Extensive experiments were performed to arrive at an optimal amount of data required for each of the languages. Starting from one hour data, increasing in steps of an hour it was observed that the system performance did not improve significantly beyond 5 hours of data. This evaluation was obtained by performing informal MOS tests. Based on the study on Tamil and Hindi, the systems for all six languages are built using five hrs of data [3]. The MOS is obtained for each language using sentences that are obtained from the web. Two different MOS scores are given in the Table 1. One is based on naturalness and the other is based on intelligibility. In the Table, the MOS for Tamil and Hindi corresponds to degradation MOS. This is reported based on the advice by [22]. Word-error-rates (WER) were obtained on semantically unpredictable sentences for Tamil and Hindi. The WER is indicated by W in the Table 1. The average number of subjects across the languages is about 30. The scores in Table 1 reveal that the MOS varies between 3.5 and 3.8 in terms of intelligibility, while it varies between 3.2 and 3.5 in terms of naturalness. Further, the MOS scores for festvox-based voices built [7] for the same languages, using 12 hrs of speech data, is also presented in Table 1 along with the MOS scores of HMM-based systems. This reiterates that HMM-based systems outperform in terms of MOS for intelligibility, while the USS system yields better results in terms of naturalness for all languages. The salient point of the work presented here is that: given the common Question Set, the only effort required to build a speech synthesis for a new language is mapping of the language's graphemes to the common phone set.

The website for the HTS system for the six languages is `http://www.iitm.ac.in/donlab/hts/`. Most languages support UTF-8.

7. Conclusion

HMM-based speech synthesis systems are built for six Indian languages, namely, Tamil, Malayalam, Telugu, Hindi, Marathi and Bengali. Owing to the phonetic similarities among Indian languages, a common phone set and a common question set are derived, thereby simplifying the task of building TTSes for Indian languages. Common acoustic models are built to hasten segmentation process. The MOS obtained for the six languages is slightly less than 3.0 in terms of naturalness, while it is higher than 3.0 in terms of intelligibility. As the phone set and question set are now generalised, a multilingual system can be developed by using adaptation techniques. Currently using the idea of the common phoneset and question set, another set of seven Indian languages TTSes in both the festival and HTS framework are at different stages of development. The success of the synthesiser does pave the way for polyglot based synthesisers for Indian languages. In particular, most Indians are trilingual (English, mother tongue and Hindi). Extending the idea of the common phone set to include languages that have

[3] At the time of this writing HMM based systems have been developed for 13 languages

Table 1: MOS for Speech Synthesised Using HTS and USS for different Indian Languages. N corresponds to naturalness, I corresponds to intelligibility, W corresponds to word error rate (in %)

Method	Tamil			Telugu		Malayalam		Marathi		Bengali		Hindi		
	N	I	W	N	I	N	I	N	I	N	I	N	I	W
HTS	2.97	3.72	6.61%	2.94	3.2	2.82	2.97	2.57	3.24	2.9	3.6	3.0	3.77	3.28%
USS	3.23	3.49	7.52%	3.01	2.65	3.33	4.1	3.84	3.58	3.56	3.59	3.597	3.602	7.02%

origins in Sino Tibetan and Austric, it should be possible to have the same voice speaking any Indian language.

Acknowledgment

The authors would like to thank the Department of Information Technology, Ministry of Communication and Technology, Government of India, for funding the project,"Development of Text-to-Speech synthesis for Indian Languages Phase II", Ref. no. 11(7)/2011-HCC(TDIL). The authors would like to thank G R Kasthuri, Asha Talambedu, Jeena Prakash and Lakshmi Priya for performing MOS tests.

8. References

[1] A. J. Hunt and A. W. Black, "Unit selection in a concatenative speech synthesis system using a large speech database," in *Proceedings of IEEE Int. Conf. Acoust., Speech, and Signal Processing*, vol. 1, 1996, pp. 373–376.

[2] H. Zen, K. Tokuda, and A. W. Black, "Statistical parametric speech synthesis," *Speech Communication*, vol. 51, pp. 1039–1064, November 2009.

[3] K. Tokuda, H. Zen, and A. W. Black, "An hmm-based speech synthesis system applied to english," in *Speech Synthesis*, 2002, pp. 227–230.

[4] A. K. Singh, "A computational phonetic model for Indian language scripts," in *In Constraints on Spelling Changes: Fifth International Workshop on Writing Systems*, 2006.

[5] "Simple 4 all," http://www.simple4all.org, Centre for Speech Technology Research, Edinburgh, 2011.

[6] wikipedia, "Languages of India," http://en.wikipedia.org/wiki/Languages_of_India, 2013.

[7] H. A. Murthy and et al., "Syllable-based speech synthesis for Indian languages and their integration with screen readers," *Speech Communication*, 2012 (submitted).

[8] S. Young, G. Evermann, M. Gales, T. Hain, D. Kershaw, X. A. Liu, G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev, and P. Woodland, *The HTK Book (for HTK Version 3.4)*, Cambridge University Engineering Department, 2002.

[9] L. Prahalad, K. Prahalad, and G. L. Madhavi, "A simple approach for building transliteration editors for Indian languages," in *Journal of Zhejiang University Science*, vol. 6A, no. 11, 2005, pp. 1354–1361.

[10] P. Eswar, "A rule based approach for spotting characters from continuous speech in Indian languages," PhD Dissertation, Indian Institute of Technology, Department of Computer Science and Eng., Madras, India, 1991.

[11] J. Latorre, K. Iwano, and S. Furui, "Polyglot synthesis using a mixture of monolingual corpora," in *icassp*, 2005, pp. 1–4.

[12] ——, "New approach to polyglot synthesis: how to speak any language with anyone's voice," in *Proceedings of the ISCA Tutorial Research Workshop on Multilingual Speech and Language Processing*, April 2006.

[13] T. Schultz, "Global phone project," http://cs.cmu.edu/ tanja/GlobalPhone/index.html, 2000.

[14] A. Black and T. Schultz, "Speaker clustering and multilingual synthesis," in *Proceedings of the ISCA Tutorial and Research Workshop on Multilingual Speech and Language Processing*, April 2006.

[15] A. Black, P. Taylor, and R. Caley, "The Festival speech synthesis system," `http://festvox.org/festival/`, 1998.

[16] S. Arora, K. K. Arora, and S.S.Agrawal, "Using syllable as a major unit for developing an efficient concatenative hindi speech synthesiser," in *Proc. of the Int. conf. SPECOM*, 2005, pp. 675–679.

[17] J. Cholin and W. J. M. Levelt, "Effects of syllable preparation and syllable frequency in speech production: Further evidence for syllabic units at a post-lexical level," *Language and Cognitive Processes*, no. 24, pp. 662–682, 2009.

[18] M. N. Rao, S. Thomas, T. Nagarajan, and H. A. Murthy, "Text-to-speech synthesis using syllable-like units," in *National Conference on Communication*, 2005, pp. 227–280.

[19] S. Thomas, M. N. Rao, H. A. Murthy, and C. S. Ramalingam, "Natural sounding speech based on syllable-like units," in *EUSIPCO, Florence, Italy*, 2006.

[20] H. A. Murthy and B. Yegnanarayana, "Group delay functions and its application to speech processing," *Sadhana*, vol. 36, no. 5, pp. 745–782, November 2011.

[21] V. K. Prasad, "Segmentation and recognition of continuous speech," PhD Dissertation, Indian Institute of Technology, Department of Computer Science and Eng., Madras, India, 2002.

[22] S. King, "Degradation MOS and word error rate for text to speech synthesis systems," private Communication.

Cross-lingual speaker adaptation based on factor analysis using bilingual speech data for HMM-based speech synthesis

Takenori Yoshimura, Kei Hashimoto, Keiichiro Oura, Yoshihiko Nankaku, and Keiichi Tokuda

Department of Scientific and Engineering Simulation, Nagoya Institute of Technology, Nagoya, Japan

Abstract

This paper proposes a cross-lingual speaker adaptation (CLSA) method based on factor analysis using bilingual speech data. A state-mapping-based method has recently been proposed for CLSA. However, the method cannot transform only speaker-dependent characteristics. Furthermore, there is no theoretical framework for adapting prosody. To solve these problems, this paper presents a CLSA framework based on factor analysis using bilingual speech data. In this proposed method, model parameters representing language-dependent acoustic features and factors representing speaker characteristics are simultaneously optimized within a unified (maximum likelihood) framework based on a single statistical model by using bilingual speech data. This simultaneous optimization is expected to deliver a better quality of synthesized speech for the desired speaker characteristics. Experimental results show that the proposed method can synthesize better speech than the state-mapping-based method.

Index Terms: cross-lingual speaker adaptation, factor analysis, HMM-based speech synthesis

1. Introduction

The advance of internationalization has rapidly increased opportunities to communicate with people who speak different languages. However, language barriers often give rise to insufficient communication due to the difficulty of fluently speaking foreign languages. To overcome language barriers, speech-to-speech translation (S2ST) systems that translate input speech into target language speech are required. S2ST typically consists of three techniques: automatic speech recognition (ASR), machine translation (MT), and text-to-speech (TTS). Conventional S2ST systems output speech with certain speaker characteristics that are unchanged even if the input speaker changes. This results in unnatural communication (i.e., the input speaker cannot be identified from the output speech). To solve this problem, cross-lingual speaker adaptation (CLSA) techniques that output speech in another language with input speaker characteristics have been developed in TTS. Hidden Markov model (HMM)-based speech synthesis systems [1, 2] are widely used for CLSA [3, 4, 5, 6, 7, 8, 9] because they can synthesize speech with various characteristics by estimating model parameters from a small amount of adaptation data, thus making them ideal for the purposes of CLSA.

Intra-lingual speaker adaptation, which is usually just called "speaker adaptation," transforms a source model into an input speaker model using a limited amount of speech data of the input speaker. Maximum likelihood linear regression (MLLR) [10, 11], which is one of the most well-known speaker adaptation techniques in HMM-based speech synthesis, can change the speaker characteristics of synthesized speech by linear transformation of the source model parameters. In this method, it is assumed that the transforms represent only the target speaker characteristics. Recently, a state-mapping-based method has been proposed as a CLSA method using MLLR [4]. This method can adapt the speaker characteristics of an output language speech by applying linear transforms in the input language to the models in the output language according to the state-mapping information. However, since the transforms include both language-dependent characteristics and speaker-dependent characteristics, this method cannot transform only speaker-dependent characteristics. In addition, the mapping information established by minimizing the Kullback-Leibler divergence (KLD) between the two states of HMMs is not guaranteed to be optimal. That is, there is no theoretical framework for adapting prosody such as rhythm and accent that have a longer dependency than one state.

To overcome these problems, we propose a CLSA method based on factor analysis using bilingual speech data[†]. In an eigenvoice method based on factor analysis [12], model parameters and factors expressing speaker characteristics in a certain language are simultaneously optimized within a unified maximum likelihood (ML) framework based on a single statistical model. In this paper, we extend the factor analysis-based eigenvoice model to the bilingual eigenvoice model. The proposed method simultaneously optimizes the model parameters for each of the input and output languages and factors by using bilingual speech data. By assuming that the factors representing speaker characteristics are common in the two languages, the approach can estimate model parameters considering the relation between the acoustic features in the two languages. In the adaptation step, the factors estimated from the adaptation data in the input language are applied to the output language directly. As a result, the proposed method can synthesize high-quality speech with the desired speaker characteristics in the output language. Furthermore, since the estimated factors that also express prosody of the input speaker is not constrained by the state structure of HMM, unlike the state-mapping-based method, the proposed method is potentially able to adapt prosody of the input speaker.

The rest of this paper is organized as follows. Section 2 describes the state-mapping-based method. Section 3 presents the intra-lingual speaker adaptation based on factor analysis. Section 4 proposes the CLSA technique based on factor analysis, and subjective listening test results are discussed in Section 5. Finally, conclusions and future work are presented in Section 6.

[†]Bilingual data is speech data uttered by persons who are able to speak two languages equally well.

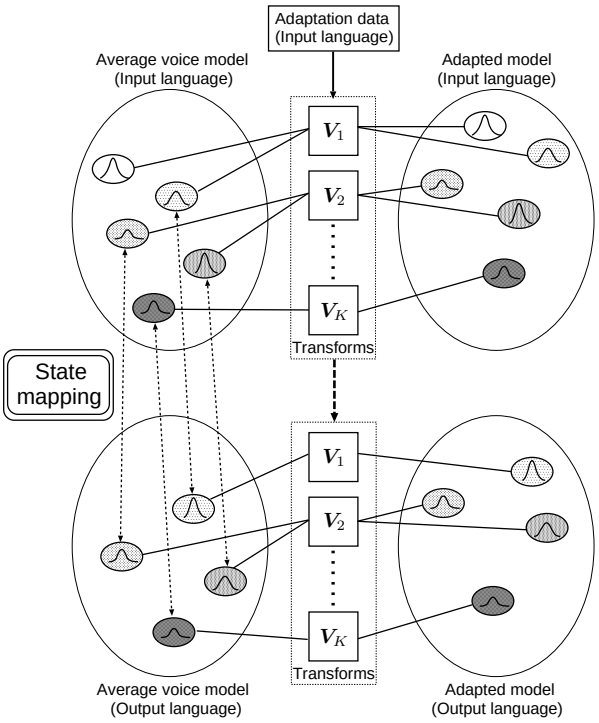


Figure 1: *Overview of the state-mapping-based method.*

2. State-mapping-based method

The basic idea of the state-mapping-based method for CLSA [4] is mapping the states of HMMs in an output language to ones in an input language. Figure 1 shows the overview of this method. First, average voice models in the input and output languages are respectively trained from speech data of various speakers by speaker adaptive training (SAT) [13]. State-mapping between these two models is then established. These mappings are estimated by searching for the state $\widehat{i}$ in the input language model that gives the minimum symmetric KLD $D_{\mathrm{KL}}(j,i)$ for each state j in the output language model:

$$\widehat{i} \;=\; \arg\min_{i} D_{\mathrm{KL}}(j,i). \tag{1}$$

Assuming that each state is represented by a single Gaussian pdf, the symmetric KLD between the two states in the input and output languages is calculated as

$$D_{\mathrm{KL}}(j,i) \;=\; D_{\mathrm{KL}}(j \,||\, i) + D_{\mathrm{KL}}(i \,||\, j), \tag{2}$$

$$D_{\mathrm{KL}}(i \,||\, j) \;=\; \frac{1}{2}\ln\left(\frac{\left|\boldsymbol{\Sigma}_j^{(O)}\right|}{\left|\boldsymbol{\Sigma}_i^{(I)}\right|}\right) + \frac{1}{2}\mathrm{Tr}\left(\boldsymbol{\Sigma}_j^{(O)^{-1}}\boldsymbol{\Sigma}_i^{(I)}\right)$$
$$-\frac{D}{2} + \frac{1}{2}\left(\boldsymbol{\mu}_j^{(O)} - \boldsymbol{\mu}_i^{(I)}\right)^{\mathsf{T}}\boldsymbol{\Sigma}_j^{(O)^{-1}}\left(\boldsymbol{\mu}_j^{(O)} - \boldsymbol{\mu}_i^{(I)}\right), \tag{3}$$

where $(\cdot)^{(I)}$ and $(\cdot)^{(O)}$ respectively represent variables in the input and output languages, $\boldsymbol{\mu}^{(\cdot)}$ and $\boldsymbol{\Sigma}^{(\cdot)}$ denote the mean vector and covariance matrix of the Gaussian pdf associated with the state indicated by its subscript, and D is the dimension of observation vector. Then, MLLR [10] or constrained MLLR (CMLLR) [11] transforms $\boldsymbol{V} = \{\boldsymbol{V}_1, \ldots, \boldsymbol{V}_K\}$ estimated from adaptation data in the input language are directly applied to the average voice models in the output language according to the established state-mapping information. By transforming all Gaussian pdfs in the output language in this way, CLSA is achieved. However, there are two theoretical problem: this method cannot transform only speaker-dependent characteristics, and cannot adapt prosody of an target speaker.

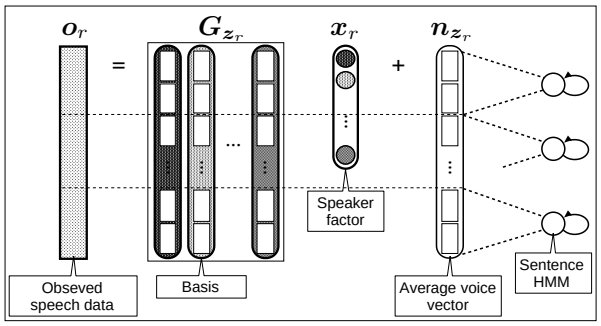


Figure 2: *Process of generating observation sequences in an eigenvoice model based on factor analysis.*

3. Speaker adaptation based on factor analysis

Eigenvoice-based methods perform speaker adaptation by changing a small number of parameters that represent various characteristics, such as speaker and speaking style. As an eigenvoice-based method, HMM-based speech synthesis using factor analysis has been proposed [12]. This approach assumes that the process of generating observation sequences is based on factor analysis, which is a statistical method for modeling the covariance structure of high-dimensional static data using a small number of latent variables. Figure 2 shows the model structure. The observation sequences $\boldsymbol{o}_r$ of speaker r are represented by

$$\boldsymbol{o}_r \;=\; \boldsymbol{G}_{\boldsymbol{z}_r}\boldsymbol{x}_r + \boldsymbol{n}_{\boldsymbol{z}_r}, \tag{4}$$

where $\boldsymbol{x}_r$, $\boldsymbol{n}_{\boldsymbol{z}_r}$, and $\boldsymbol{G}_{\boldsymbol{z}_r}$ denote the factor, noise vector, and loading matrix, respectively. This model can express various speaker characteristics by changing the factor $\boldsymbol{x}_r$. Moreover, since the noise vector $\boldsymbol{n}_{\boldsymbol{z}_r}$ and loading matrix $\boldsymbol{G}_{\boldsymbol{z}_r}$ are generated stochastically according to state sequence $\boldsymbol{z}_r$, the model can represent variable-length observation directly.

The factor $\boldsymbol{x}_r$ and noise vector $\boldsymbol{n}_r$ are often given by the following Gaussian distribution:

$$\boldsymbol{x}_r \;\sim\; \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}), \tag{5}$$
$$\boldsymbol{n}_{\boldsymbol{z}_r} \;\sim\; \mathcal{N}(\boldsymbol{\mu}_{\boldsymbol{z}_r}, \boldsymbol{\Sigma}_{\boldsymbol{z}_r}). \tag{6}$$

Under this condition, the likelihood function for acoustic feature sequences $\boldsymbol{o}$ of all speakers $(r = 1, 2, \ldots, R)$ is written as

$$P(\boldsymbol{o} \mid \boldsymbol{\Lambda}) \;=\; \prod_{r=1}^{R}\sum_{\boldsymbol{z}_r}\int P(\boldsymbol{o}_r, \boldsymbol{z}_r, \boldsymbol{x}_r \mid \boldsymbol{\Lambda})\, d\boldsymbol{x}_r$$
$$= \prod_{r=1}^{R}\sum_{\boldsymbol{z}_r}\int P(\boldsymbol{o}_r \mid \boldsymbol{z}_r, \boldsymbol{x}_r, \boldsymbol{\Lambda}) P(\boldsymbol{z}_r \mid \boldsymbol{\Lambda})$$
$$\times P(\boldsymbol{x}_r)\, d\boldsymbol{x}_r, \tag{7}$$
$$P(\boldsymbol{o}_r \mid \boldsymbol{z}_r, \boldsymbol{x}_r, \boldsymbol{\Lambda}) \;=\; \mathcal{N}(\boldsymbol{o}_r \mid \boldsymbol{G}_{\boldsymbol{z}_r}\boldsymbol{x}_r + \boldsymbol{\mu}_{\boldsymbol{z}_r}, \boldsymbol{\Sigma}_{\boldsymbol{z}_r}), \tag{8}$$

where $\mathbf{\Lambda}$ is a set of model parameters that includes the loading matrix $\boldsymbol{G}_{\boldsymbol{z}_r}$, the noise mean vector $\boldsymbol{\mu}_{\boldsymbol{z}_r}$, and the noise covariance matrix $\boldsymbol{\Sigma}_{\boldsymbol{z}_r}$. These parameters and factors are simultaneously optimized within a unified ML framework. However, the parameter estimation is computationally intractable due to the combination of latent variables: factor and state sequence. Hence, the distribution of these variables is computed via the variational expectation-maximization (VEM) algorithm [12], which is an iterative algorithm that is closely related to the standard expectation-maximization (EM) algorithm. For speaker adaptation, the factor $\boldsymbol{x}_{r'}$ is estimated using the trained model and adaptation data of the target speaker r'. The estimated factor $\boldsymbol{x}_{r'}$ represents the characteristics of the target speaker r' so that synthesized speech with the characteristics of speaker r' can be obtained.

4. Cross-lingual speaker adaptation based on factor analysis

In the eigenvoice method based on factor analysis [12] described above, the model parameters and factors expressing speaker characteristics in a certain language are simultaneously optimized within a unified ML framework based on a single statistical model. In this paper, we extend the factor analysis-based eigenvoice model to the bilingual eigenvoice model. The proposed method simultaneously optimizes the model parameters for each of the input and output languages and factors by using bilingual speech data. By assuming that the factors representing speaker characteristics are common in the two language, the proposed method can estimate model parameters considering the relation between acoustic features in the two languages. In the adaptation step, the factors estimated using adaptation data in the input language are applied to the output language directly. Therefore, the same speaker characteristics can be obtained in another language. Furthermore, since the estimated factors that also express prosody of the input speaker is not influenced by language-dependent information such as context or the state structure of HMM, the proposed method can adapt the prosody of the input speaker in principle.

4.1. Model structure based on factor analysis in multilingual space

Figure 3 gives the model structure of the proposed method. Bilingual data $\boldsymbol{o}$, which consists of input language data $\boldsymbol{o}^{(I)}$ and output language data $\boldsymbol{o}^{(O)}$ uttered by bilingual speakers $(r = 1, 2, \ldots, R)$, is used for training the proposed model. The likelihood function is calculated as

$$
\begin{aligned}
P(\boldsymbol{o} \mid \mathbf{\Lambda}) &= \prod_{r=1}^{R} \int P\left(\boldsymbol{o}_r^{(O)}, \boldsymbol{o}_r^{(I)}, \boldsymbol{x}_r \,\middle|\, \mathbf{\Lambda}^{(O)}, \mathbf{\Lambda}^{(I)}\right) d\boldsymbol{x}_r \\
&= \prod_{r=1}^{R} \int P\left(\boldsymbol{o}_r^{(O)} \,\middle|\, \boldsymbol{x}_r, \mathbf{\Lambda}^{(O)}\right) P\left(\boldsymbol{o}_r^{(I)} \,\middle|\, \boldsymbol{x}_r, \mathbf{\Lambda}^{(I)}\right) \\
&\quad \times P\left(\boldsymbol{x}_r\right) d\boldsymbol{x}_r, \qquad (9)
\end{aligned}
$$

where $\mathbf{\Lambda}$ is composed of sets of model parameters for the input and output languages, $\mathbf{\Lambda}^{(I)}$ and $\mathbf{\Lambda}^{(O)}$. A set of model parameters $\mathbf{\Lambda}^{(\cdot)}$ includes the loading matrix $\boldsymbol{G}^{(\cdot)}$, the noise mean vector $\boldsymbol{\mu}^{(\cdot)}$, and the noise covariance matrix $\boldsymbol{\Sigma}^{(\cdot)}$. The factor $\boldsymbol{x}_r$ representing speaker characteristics is independent from languages, i.e., the common speaker factor is used in the input and output languages, as shown in Fig. 3. In Eq. (9),

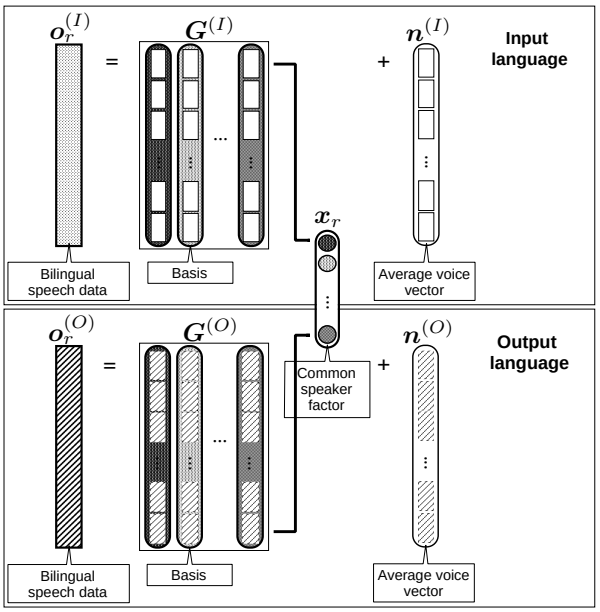


Figure 3: *Model structure of cross-lingual speaker adaptation based on factor analysis using bilingual speech data.*

$P\left(\boldsymbol{o}_r^{(I)} \,\middle|\, \boldsymbol{x}_r, \mathbf{\Lambda}^{(I)}\right)$ and $P\left(\boldsymbol{o}_r^{(O)} \,\middle|\, \boldsymbol{x}_r, \mathbf{\Lambda}^{(O)}\right)$ denote the output probability of the input and output language, respectively. This approach can simultaneously model the speech data of different languages. Therefore, the model parameters $\mathbf{\Lambda}^{(I)}$ and $\mathbf{\Lambda}^{(O)}$ are trained by a similar algorithm to the one using monolingual data (Eq. (7)) in the training step. Note that state sequence is omitted from above.

4.2. Variational expectation-maximization algorithm

A lower bound $\mathcal{F}$ of the log likelihood is defined by using Jensen's inequality:

$$
\begin{aligned}
&\log P(\boldsymbol{o} \mid \mathbf{\Lambda}) \\
&= \log \prod_{r=1}^{R} \sum_{\boldsymbol{z}_r^{(O)}, \boldsymbol{z}_r^{(I)}} \\
&\quad \int P\left(\boldsymbol{o}_r^{(O)}, \boldsymbol{z}_r^{(O)}, \boldsymbol{o}_r^{(I)}, \boldsymbol{z}_r^{(I)}, \boldsymbol{x}_r \,\middle|\, \mathbf{\Lambda}\right) d\boldsymbol{x}_r \\
&\geq \sum_{r=1}^{R} \sum_{\boldsymbol{z}_r^{(O)}, \boldsymbol{z}_r^{(I)}} \int Q\left(\boldsymbol{z}_r^{(O)}, \boldsymbol{z}_r^{(I)}, \boldsymbol{x}_r\right) \\
&\quad \times \log \frac{P\left(\boldsymbol{o}_r^{(O)}, \boldsymbol{z}_r^{(O)}, \boldsymbol{o}_r^{(I)}, \boldsymbol{z}_r^{(I)}, \boldsymbol{x}_r \,\middle|\, \mathbf{\Lambda}\right)}{Q\left(\boldsymbol{z}_r^{(O)}, \boldsymbol{z}_r^{(I)}, \boldsymbol{x}_r\right)} d\boldsymbol{x}_r \\
&\equiv \mathcal{F}, \qquad (10)
\end{aligned}
$$

where $\boldsymbol{z}_r^{(\cdot)}$ is a state sequence and $Q\left(\boldsymbol{z}_r^{(O)}, \boldsymbol{z}_r^{(I)}, \boldsymbol{x}_r\right)$ is an arbitrary function. Then, the relation between the log likelihood and the lower bound $\mathcal{F}$ is represented as

$$
\mathcal{F} = \log P(\boldsymbol{o} \mid \mathbf{\Lambda}) - D_{\mathrm{KL}}(Q \,||\, P). \qquad (11)
$$

Therefore, maximizing the lower bound $\mathcal{F}$ is equivalent to minimizing the KLD $D_{\mathrm{KL}}(Q \,||\, P)$. The arbitrary function

$Q\left(\boldsymbol{z}_r^{(O)}, \boldsymbol{z}_r^{(I)}, \boldsymbol{x}_r\right)$ is then regarded as an approximate posterior distribution because $Q\left(\boldsymbol{z}_r^{(O)}, \boldsymbol{z}_r^{(I)}, \boldsymbol{x}_r\right)$ approximates the true posterior distribution $P\left(\boldsymbol{z}_r^{(O)}, \boldsymbol{z}_r^{(I)}, \boldsymbol{x}_r \,\middle|\, \boldsymbol{o}_r^{(O)}, \boldsymbol{o}_r^{(I)}, \boldsymbol{\Lambda}\right)$ when $D_{\mathrm{KL}}(Q \mid\mid P)$ is reduced. The approximate posterior distribution $Q\left(\boldsymbol{z}_r^{(O)}, \boldsymbol{z}_r^{(I)}, \boldsymbol{x}_r\right)$ is estimated by maximizing the lower bound $\mathcal{F}$. However, it is difficult to estimate $Q\left(\boldsymbol{z}_r^{(O)}, \boldsymbol{z}_r^{(I)}, \boldsymbol{x}_r\right)$ directly. To compute it easily, the VEM method assumes that the latent variables are independent under the condition that observation sequences $\boldsymbol{o}_r$ is given:

$$Q\left(\boldsymbol{z}_r^{(O)}, \boldsymbol{z}_r^{(I)}, \boldsymbol{x}_r\right) = Q\left(\boldsymbol{z}_r^{(O)}\right) Q\left(\boldsymbol{z}_r^{(I)}\right) Q(\boldsymbol{x}_r). \quad (12)$$

Under this assumption, the optimal posterior distributions can be computed by a similar procedure to the EM algorithm, which increases the value of the objective function $\mathcal{F}$ at each iteration until convergence. The posterior distributions of state sequences $Q\left(\boldsymbol{z}_r^{(\cdot)}\right)$ and factor $Q(\boldsymbol{x}_r)$ are obtained by adapting the variational method to the lower bound $\mathcal{F}$:

$$Q\left(\boldsymbol{z}_r^{(O)}\right) = C_{\boldsymbol{z}_r^{(O)}} P\left(\boldsymbol{z}_r^{(O)} \,\middle|\, \boldsymbol{\Lambda}^{(O)}\right) \exp\left[\int Q(\boldsymbol{x}_r) \times \log P\left(\boldsymbol{o}_r^{(O)} \,\middle|\, \boldsymbol{z}_r^{(O)}, \boldsymbol{x}_r, \boldsymbol{\Lambda}^{(O)}\right) d\boldsymbol{x}_r\right], \quad (13)$$

$$Q\left(\boldsymbol{z}_r^{(I)}\right) = C_{\boldsymbol{z}_r^{(I)}} P\left(\boldsymbol{z}_r^{(I)} \,\middle|\, \boldsymbol{\Lambda}^{(I)}\right) \exp\left[\int Q(\boldsymbol{x}_r) \times \log P\left(\boldsymbol{o}_r^{(I)} \,\middle|\, \boldsymbol{z}_r^{(I)}, \boldsymbol{x}_r, \boldsymbol{\Lambda}^{(I)}\right) d\boldsymbol{x}_r\right], \quad (14)$$

$$Q(\boldsymbol{x}_r) = C_{\boldsymbol{x}_r} P(\boldsymbol{x}_r) \exp\left[\sum_{\boldsymbol{z}_r^{(O)}} Q\left(\boldsymbol{z}_r^{(O)}\right) \log P\left(\boldsymbol{o}_r^{(O)} \,\middle|\, \boldsymbol{z}_r^{(O)}, \boldsymbol{x}_r, \boldsymbol{\Lambda}^{(O)}\right) + \sum_{\boldsymbol{z}_r^{(I)}} Q\left(\boldsymbol{z}_r^{(I)}\right) \log P\left(\boldsymbol{o}_r^{(I)} \,\middle|\, \boldsymbol{z}_r^{(I)}, \boldsymbol{x}_r, \boldsymbol{\Lambda}^{(I)}\right)\right], \quad (15)$$

where $C_{\boldsymbol{z}_r^{(I)}}$, $C_{\boldsymbol{z}_r^{(O)}}$, and $C_{\boldsymbol{x}_r}$ are the normalization terms to ensure $\sum_{\boldsymbol{z}_r^{(\cdot)}} Q\left(\boldsymbol{z}_r^{(\cdot)}\right) = 1$ and $\int Q\left(\boldsymbol{x}_r\right) d\boldsymbol{x}_r = 1$. Since the posterior distributions depend on each other, they should be optimized simultaneously by iterative procedures. If the factor $\boldsymbol{x}_r$ and the noise vector are given by Gaussian distributions as Eqs. (5) and (6), the posterior distribution $Q\left(\boldsymbol{x}_r\right)$ is expressed as

$$Q\left(\boldsymbol{x}_r\right) = \mathcal{N}\left(\boldsymbol{x}_r \,\middle|\, \hat{\boldsymbol{\mu}}_{\boldsymbol{x}_r}, \hat{\boldsymbol{\Sigma}}_{\boldsymbol{x}_r}\right), \quad (16)$$

where $\hat{\boldsymbol{\mu}}_{\boldsymbol{x}_r}$ and $\hat{\boldsymbol{\Sigma}}_{\boldsymbol{x}_r}$ denote the mean vector and covariance matrix of factor $\boldsymbol{x}_r$, respectively.

4.3. Adaptation step

In the adaptation step (Fig. 4), the optimal speech parameter sequence in the output language $\hat{\boldsymbol{o}}_{r'}^{(O)}$ of an input speaker r', who is not a bilingual speaker, is generated by maximizing the output probability:

$$\hat{\boldsymbol{o}}_{r'}^{(O)} = \arg\max_{\boldsymbol{o}_{r'}^{(O)}} \int P\left(\boldsymbol{o}_{r'}^{(O)}, \boldsymbol{o}_{r'}^{(I)}, \boldsymbol{x}_{r'} \,\middle|\, \boldsymbol{\Lambda}^{(O)}, \boldsymbol{\Lambda}^{(I)}\right) d\boldsymbol{x}_{r'} = \arg\max_{\boldsymbol{o}_{r'}^{(O)}} \int P\left(\boldsymbol{o}_{r'}^{(O)} \,\middle|\, \boldsymbol{x}_{r'}, \boldsymbol{\Lambda}^{(O)}\right) \times P\left(\boldsymbol{x}_{r'} \,\middle|\, \boldsymbol{o}_{r'}^{(I)}, \boldsymbol{\Lambda}^{(I)}\right) d\boldsymbol{x}_{r'}. \quad (17)$$

It is difficult to estimate the true posterior distribution of the factor $\boldsymbol{x}_{r'}$ because the likelihood function includes multiple latent variables. Therefore, the same approximation as the training step, i.e., the VEM algorithm, is applied to estimate the approximate posterior distribution of $\boldsymbol{x}_{r'}$. In addition, the maximum a posterior approximation is applied. Consequently, the optimal speech parameter sequence is generated by using the mean vector of the posterior distribution $\hat{\boldsymbol{\mu}}_{\boldsymbol{x}_{r'}}$, which obtains the maximum posterior probability, as

$$\hat{\boldsymbol{o}}_{r'}^{(O)} \approx \arg\max_{\boldsymbol{o}_{r'}^{(O)}} \int P\left(\boldsymbol{o}_{r'}^{(O)} \,\middle|\, \boldsymbol{x}_{r'}, \boldsymbol{\Lambda}^{(O)}\right) Q\left(\boldsymbol{x}_{r'}\right) d\boldsymbol{x}_{r'} \approx \arg\max_{\boldsymbol{o}_{r'}^{(O)}} P\left(\boldsymbol{o}_{r'}^{(O)} \,\middle|\, \hat{\boldsymbol{\mu}}_{\boldsymbol{x}_{r'}}, \boldsymbol{\Lambda}^{(O)}\right). \quad (18)$$

From this equation, the estimated factor is immediately applied to the output probability distribution in the output language. As a result, the synthesized speech does not depend on the input language-dependent characteristics, so that the high-performance sound is expected.

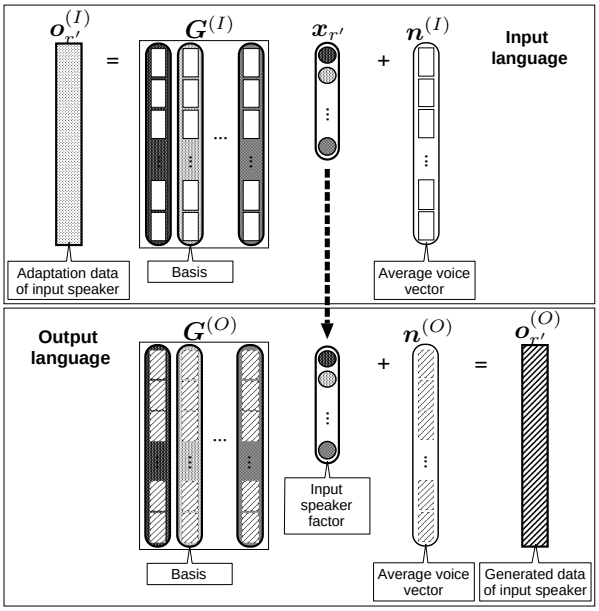


Figure 4: *Overview of the adaptation step in cross-lingual speaker adaptation based on factor analysis using bilingual speech data.*

4.4. Cross-lingual speaker adaptation based on speaker interpolation

In the proposed framework, CLSA based on speaker interpolation can be represented. This method replaces bases of loading matrices $\boldsymbol{G}^{(\cdot)}$ with the mean vectors $\bar{\boldsymbol{\mu}}_r^{(\cdot)}$ of the speaker-dependent model of each training speaker:

$$\boldsymbol{G}^{(I)} = \left[\begin{array}{cccc} \bar{\boldsymbol{\mu}}_1^{(I)} & \bar{\boldsymbol{\mu}}_2^{(I)} & \cdots & \bar{\boldsymbol{\mu}}_R^{(I)} \end{array} \right], \quad (19)$$

$$\boldsymbol{G}^{(O)} = \left[\begin{array}{cccc} \bar{\boldsymbol{\mu}}_1^{(O)} & \bar{\boldsymbol{\mu}}_2^{(O)} & \cdots & \bar{\boldsymbol{\mu}}_R^{(O)} \end{array} \right], \quad (20)$$

and substitutes noise mean vectors $\boldsymbol{\mu}^{(\cdot)}$ with zero vectors:

$$\boldsymbol{\mu}^{(I)} = \boldsymbol{\mu}^{(O)} = \boldsymbol{0}. \quad (21)$$

Since the training speaker models are weighted by the factor $\boldsymbol{x}_{r'}$ of the input speaker r', this method is regarded as speaker interpolation. By applying the factor $\boldsymbol{x}_{r'}$ estimated from the adaptation data $\boldsymbol{o}_{r'}^{(I)}$ of the input speaker r' to the output language directly, speaker interpolation in the output language is achieved. Hence, the method is considered an approach that approximates ML criterion based optimization in the proposed method. To evaluate the effectiveness of model parameter optimization in the proposed method, we included this interpolation approach in the following experiments.

5. Experiments

5.1. Experimental setups

We used a Japanese-English and Japanese-Chinese bilingual monologue speech database [14] in these experiments. The speech signals were sampled at a rate of 48 kHz and windowed by a 25-ms Hamming window with a 5-ms shift. Feature vectors consisted of 34 mel-cepstral coefficients including the zeroth coefficient, fundamental frequency (F_0), and their first and second time derivatives. A five-state left-to-right no-skip context-dependent MSD-HSMM [15, 16] was used. The input language was English and the output language was Japanese. For training the models, 2,700 sentences uttered by five female speakers were used in each language. Two English sentences were used as adaptation data. The target speakers were four female speakers who were not included in the training speakers.

Two subjective listening tests were conducted. The first test evaluated the naturalness of the synthesized speech by the mean opinion score (MOS) test method, and the second one evaluated the speaker similarity between the target speech and the synthesized speech for each model by the differential MOS (DMOS) test method. In the MOS test, after the subjects had listened to a test sample, they were asked to assign it a naturalness score on a five-point scale (5: excellent, 4: good, 3: fair, 2: poor, 1: bad). In the DMOS test, after the subjects had listened to Japanese natural speech of the target speaker and a test sample, they were asked to assign it a similarity score on a five-point scale (5: very close, 4: close, 3: fair, 2: far, 1: very far). Ten subjects evaluated 15 and 12 sentences, which were randomly chosen from 45 sentences, in the MOS and DMOS tests, respectively.

The following methods were compared.

- **SD**: The speaker-dependent model of each target speaker.
- **SM**: CLSA based on state-mapping (conventional method).
- **IP**: CLSA based on speaker interpolation (approximation method).
- **FA**: CLSA based on factor analysis (proposed method).

Table 1: *The number of sentences for training the speaker-dependent model.*

Speaker	EJF04	EJF06	EJF10	EJF11
No. of sentences	1400	1200	167	153

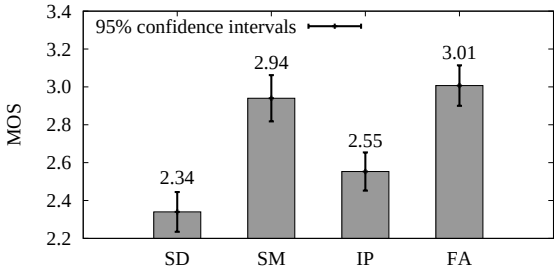


Figure 5: *Experimental results (naturalness).*

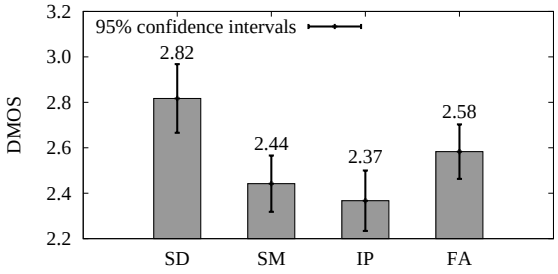


Figure 6: *Experimental results (speaker similarity).*

The number of sentences used for training the **SD** models is listed in Table 1. The parameters of the spectral, F_0, and duration models were adapted to the target speaker in **IP** and **FA**, but adaptation of durations was not performed in **SM** because it is considered unsuitable [4]. For constructing **IP** models using Eqs. (19), (20), and (21), speaker-dependent models of training speakers were constructed by applying CMLLR transformations for each training speaker to average voice models. The average voice models for the input and output languages were the same as the ones used in **SM**. The number of bases for **FA** was three in these experiments.

5.2. Experimental results

The results of the MOS and DMOS listening tests are shown in Figs. 5 and 6, respectively, with the vertical line indicating the 95% confidence intervals. Figure 5 shows that **FA** significantly improved the naturalness of the synthesized speech compared with **SD**. This is because **FA** can estimate appropriate acoustic models by efficiently using training data of various speakers while **SD** uses the data of only the target speaker. **IP** also obtained a higher score than **SD**. However, **FA** achieved a greater improvement than **IP**, indicating that **FA** can estimate the bases and noise vector properly and that simultaneous op-

Table 2: *DMOS for each target speaker.*

	SD	SM	IP	FA
EJF04	2.52	2.43	2.81	2.38
EJF06	3.14	2.71	2.60	2.54
EJF10	2.90	2.35	2.00	2.77
EJF11	2.58	2.24	2.18	2.58
Mean	2.82	2.44	2.37	2.58

timization based on the proposed method is effective. **FA** and **SM** had similar results. **SM** often occurred discontinuity errors due to not considering the contextual factors in the output language. In contrast, **FA** generated smooth synthesized speech, although the naturalness of the speech was often degraded by over-smoothing. To overcome the over-smoothing problem, a training algorithm for the proposed method will be investigated in the future.

It can be seen from Fig. 6 that **FA** delivered suitable speaker similarity. **FA** obtained a higher score than **SM**, i.e., **FA** synthesized speech with more similar speaker characteristics to the target speaker ones than **SM**. Furthermore, **FA** outperformed **IP**, although **FA** had a smaller number of bases than **IP**. These results suggest that the proposed model can represent various speaker characteristics and that simultaneous optimization is effective to estimate the appropriate model parameters in terms of representation of speaker variations. Table 2 gives the detailed DMOS results for each target speaker. From Tables 1 and 2, **FA** showed a nearly identical performance to **SD** in the case of speaker-dependent models trained by a small amount of training data, such as EJF10 and EJF11. This indicates that the proposed method has potential for delivering more suitable speaker similarity by investigating the model structure such as the number of bases.

6. Conclusion

We proposed a cross-lingual speaker adaptation method based on factor analysis using bilingual speech data for HMM-based speech synthesis. In the proposed method, model parameters representing language-dependent acoustic features and factors representing speaker characteristics are simultaneously optimized within a unified framework based on a single statistical model by using bilingual speech data. The results of subjective listening tests indicated that the proposed method can generate natural synthesized speech with suitable speaker similarity. In addition, the effectiveness of the simultaneous optimization was shown by the experimental results. Our future work is to investigate the model structure, such as the number of bases and the utilization of monolingual speech data.

7. Acknowledgments

This research was partly funded by Core Research for Evolutional Science and Technology (CREST) from Japan Science and Technology Agency (JST).

8. References

[1] T. Yoshimura, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, "Simultaneous modeling of spectrum, pitch and duration in HMM-based speech synthesis," *Proc. of Eurospeech 1999*, pp. 2347–2350, 1999.

[2] K. Tokuda, T. Yoshimura, T. Masuko, T. Kobayashi, and T. Kitamura, "Speech parameter generation algorithms for HMM-based speech synthesis," *Proc. of ICASSP 2000*, vol. 3, pp. 1315–1318, 2000.

[3] Y. J. Wu, S. King, and K. Tokuda, "Cross-Lingual speaker adaptation for HMM-based speech synthesis," *Proc. of ISCSLP 2008*, pp. 1–4, 2008.

[4] Y. J. Wu, Y. Nankaku, and K. Tokuda, "State mapping based method for cross-lingual speaker adaptation in HMM-based speech synthesis," *Proc. of Interspeech 2009*, pp. 528–531, 2009.

[5] K. Oura, J. Yamagishi, M. Wester, S. King, and K. Tokuda, "Unsupervised cross-lingual speaker adaptation for speech-to-speech translation system," *Proc. of ICASSP 2010*, pp. 4594–4597, 2010.

[6] M. Gibson, T. Hirsimäki, R. Karhila, M. Kurimo, and W. Byrne, "Unsupervised cross-lingual speaker adaptation for HMM-based speech synthesis using two-pass decision tree construction," *Proc. of ICASSP 2010*, pp. 4642–4645, 2010

[7] X. Peng, K. Oura, Y. Nankaku, and, K. Tokuda, "Cross-lingual speaker adaptation for HMM-based speech synthesis considering differences between language-dependent average voices," *Proc. of ICSP 2010*, pp. 605–608, 2010.

[8] N. Hattori, T. Toda, H. Kawai, H. Saruwatari, and K. Shikano, "Speaker-adaptive speech synthesis based on eigenvoice conversion and language-dependent prosodic conversion in speech-to-speech translation," *Proc. of Interspeech 2011*, pp. 2769–2772, 2011.

[9] H. Zen, N. Braunschweiler, S. Buchholz, M. J. F. Gales, K. Knill, S. Krstulović, and J. Latorre, "Statistical parametric speech synthesis based on speaker and language factorization," *IEEE Trans.*, vol. 20, no. 6, pp. 1713–1724, 2012.

[10] C. J. Leggetter and P. C. Woodland, "Maximum likelihood linear regression for speaker adaptation of continuous density hidden Markov models," *Computer Speech and Language*, vol. 9, pp. 171–185, 1995.

[11] M. J. F. Gales, "Maximum likelihood linear transformations for HMM-based speech recognition," *Computer Speech and Language*, vol. 12, no. 2, pp. 75–98, 1998.

[12] K. Kazumi, Y. Nankaku, and K. Tokuda, "Factor analyzed voice models for HMM-based speech synthesis," *Proc. of ICASSP 2010*, pp. 4234–4237, 2010.

[13] J. Yamagishi and T. Kobayashi, "Average-voice-based speech synthesis using HSMM-based speaker adaptation and adaptive training," *IEICE Trans.*, vol. E90–D, no. 2, pp. 533–543, 2007.

[14] ALAGIN language/voice resources site, Online: https://alaginrc.nict.go.jp/resources/nictmastar/menuspeechlist/speechoutline.html, accessed on 2 Mar 2013.

[15] K. Tokuda, T. Masuko, N. Miyazaki, and T. Kobayashi, "Hidden Markov models based on multi-space probability distribution for pitch pattern modeling," *Proc. of ICASSP 1999*, vol. 1, pp. 229–232, 1999.

[16] H. Zen, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, "A hidden semi-Markov model-based speech synthesis system," *IEICE Trans.*, vol. E90–D, no. 5, pp. 825–834, 2007.

Residual Compensation based on Articulatory Feature-based Phone Clustering for Hybrid Mandarin Speech Synthesis

Yi-Chin Huang, Chung-Hsien Wu, Shih-Lun Lin

Department of Computer Science and Information Engineering
National Cheng Kung University, Tainan, TAIWAN
ychin.huang@gmail.com, chunghsienwu@gmail.com

Abstract

While speech synthesis based on Hidden Markov Models (HMMs) has been developed to successfully synthesize stable and intelligible speech with flexibility and small footprints in recent years, HMM-based method is still incapable to generate the speech with good quality and high naturalness. In this study, a hybrid method combining the unit-selection and HMM-based methods is proposed to compensate the residuals between the feature vectors of the natural phone units and the HMM-synthesized phone units to select better units and improve the naturalness of the synthesized speech. Articulatory features are adopted to cluster the phone units with similar articulation to construct the residual models of phone clusters. One residual model is characterized for each phone cluster using state-level linear regression. The candidate phone units of the natural corpus are selected by considering the compensated synthesized phone units of the same phone cluster, and then an optimal phone sequence is decided by the spectral features, contextual articulatory features, and pitch values to generate the synthesized speech with better naturalness. Objective and subjective evaluations were conducted and the comparison results to the HMM-based method and the conventional hybrid-based method confirm the improved performance of the proposed method.
Index Terms: Articulatory Feature, HMM-based TTS, Hybrid method, Residual Model, Unit Selection

1. Introduction

The performance of speech synthesis for unit selection-based [1] [2] and statistical method, especially Hidden Markov Model (HMM)-based methods [3], has been improved considerably in recent years. With its flexibility to convert or adapt the synthesized speech to different speakers [4], languages [5] [6], or even emotions [7], the statistical speech synthesis method generally requires less effort on recording speech utterances when compared to the unit selection-based method. On the contrary, even though unit-selection-based synthesis method requires a very large speech database to obtain various instances of phonetic or prosodic information, the resultant synthesized speech can have much better voice quality than the statistical TTS systems. In recent years, many researchers have focused on combining both methods and developing a hybrid approach to obtain the synthesized speech with high quality. [8] [9] [10] The main research interest of the hybrid method is to calculate the target cost based on the synthesized speech from the statistical synthesis method. Spectral parameters, F_0 values, and duration information are generated from the HMM-based TTS system as the "targets" for unit selection. Generally, these studies used maximum-likelihood (ML)-estimated HMMs to predict the targets or calculate the costs between natural phone units and the corresponding synthesized phone units. In [11], a hybrid approach called minimum unit-selection error (MUSE) training was proposed. In this approach, the number of different units between the selected units and natural units define the loss function. Then, HMM parameters are optimized to minimize the loss function using the generalized probabilistic descent (GPD) method.

In the frameworks of the hybrid systems which combine the statistical synthesis method and unit-selection-based method, the linguistic information and the output speech of statistical-based synthesizer are usually used to further select appropriate synthesis unit in the unit-selection method. However, the output speech of the statistical-based synthesizer usually tends to be over-smoothing than the original speech unit. Figure 1 shows an example of a natural Mandarin phone unit /shu/ and its synthesized unit generated by the HMM-based method with forced alignment to the natural phone. As the figure shows, there is an essential difference between the synthesized unit and the natural one because the HMM-based method suffers from the over-smoothing problem due to its statistics nature. Therefore, the residual between the synthesized speech and the natural speech should be considered in the hybrid-based method and it could be used to compensate the synthesized speech for better unit selection. In this study, we investigated the differences between natural speech and synthesized speech to construct the residual models based on linguistic and articulatory information of the phone units, and proposed a novel approach to residual compensation for selecting more suitable speech units for hybrid Mandarin speech synthesis.

The rest of the paper is organized as follows. The proposed system framework of hybrid speech synthesis system is introduced in Section 2, consisting of phone clustering based on articulatory features and the resultant average residual model of each phone cluster. Section 3 consists of Subjective and objective evaluations of the proposed system comparing with the original HMM-based TTS system. Lastly, concluding remarks are given in Section 4.

2. System Description

In this section, the proposed hybrid Mandarin speech synthesis system is introduced. Phone clustering based on the articulatory features, state-based average residual models construction for compensating the differences between natural speech and synthesized speech, and the synthesis phase of the Mandarin speech synthesis system are presented.

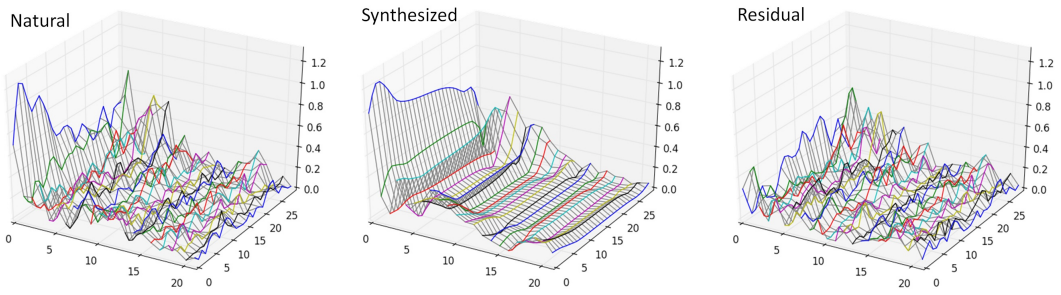


Figure 1: An example of residual MGC coefficient of natural phone /Shu/ and its synthesized one

Table 1: Articulatory Attributes used for Mandarin.

Vowel-related			
vowel	Front	Back	High
Medium	Low	Round	Dipthong
A-vowel	E-vowel	I-vowel	O-vowel
U-vowel	R-vowel		
Consonant-related			
Consonant	Front	Back	Dental
Retroflex	Velar	Plosive	Fricative
Approximant	Lateral	Affricate	Liquid
Labial	Coronal		
Mix			
Ax	W	Y	Continuant
Nasal	Voiced		
Pause			
Pause			

2.1. Phone Clustering based on Articulatory Features

Articulatory attributes, which are speaker-independent and language-independent, are useful in speech processing, especially in dealing with pronunciation variation. [12] [13] [14]

Each speech frame can be modeled by a vector in the space spanned by the articulatory attributes and a speech segment could be a feature contour in the space of the articulatory attributes. In this research, an artificial neural network (ANN) for different articulatory attributes is constructed to extract the articulatory features. The input layer consists of nodes for the static and dynamic mel-general cepstral (MGC) coefficients. The two-level hidden layers, each consisting of 30 nodes, is used based on the preliminary experimental results. The nodes at the output layer represent the posterior probabilities of 35 articulatory attributes in Mandarin speech. The articulatory attributes are listed in Table 1. The output values, which are defined as articulatory features, are the likelihoods of the articulatory attributes.

In this study, the articulatory features of the training data are extracted and used for phone clustering. Clustering of phones in the speech corpus, based on the similarity of acoustic features rather than linguistic features, is performed. In this case, we can further investigate the relationship between the natural speech and its corresponding synthesized speech in the same cluster. The procedure is described as follows. First, the phone identity and phone duration are considered and can be used to investigate the distributions of articulatory features occur in the same phone cluster with similar speaking rate. Then, the K-means clustering method based on the Euclidean distance of articulatory features between phones is adopted.

Finally, the phone clusters obtained based on articulatory features are obtained, and the number of phones in each cluster is set to be smaller than 30 based on the experimental results.

2.2. Average Residual Models

The research goal in this study is to compensate the residual between the natural and the synthesized speech, and use the compensated synthesized speech to select appropriate units for natural speech synthesis. To investigate the relationship between natural phones and synthesized phones in the same phone cluster, we trained the HMM-based synthesis models using the same speech corpus for phone clustering, and then the synthesized speech is aligned to the original natural speech based on the state information. In this way, we can calculate the differences of MGC coefficients between each phone pair of natural and synthesized phones at the state level. Linear interpolation is adopted to normalize the difference of phone length. For each cluster, the differences of states are then used to calculate the regression line to represent the average residual model of each state for the phone cluster.

To validate the effectiveness of the average residual model for each cluster, the Mean Cepstral Distance (MCD) is used to evaluate whether the compensated synthesized speech is more similar to the natural speech with respect to cepstral distance. The MCD is calculated below:

$$MCD = \frac{1}{N} \sum_{n=0}^{N-1} (y_n - \tilde{y}_n)^2 \quad (1)$$

where N is the total number of MGC feature vectors. y_n and $\tilde{y}_n$ denote the natural and the compensated feature vectors. The average residual model is aligned to the synthesized speech based on state information. The results show that 90.9% of the synthesized phones achieved smaller MCD when the average residual model is used for compensation, and the total MCD decreased for the inside test. This result shows that the proposed residual model is useful to alleviate the over-smoothing problem between natural and synthesized speech.

2.2.1. Synthesis Phase

There are two parts in the synthesis phase. In the first part, the synthesized phone unit are compensated by the average residual models, as Fig. 2 shows. In the second part, the compensated synthesized units are used to expand similar synthesis units of natural speech for selecting suitable units to generate the speech with larger dynamic range of acoustic features.

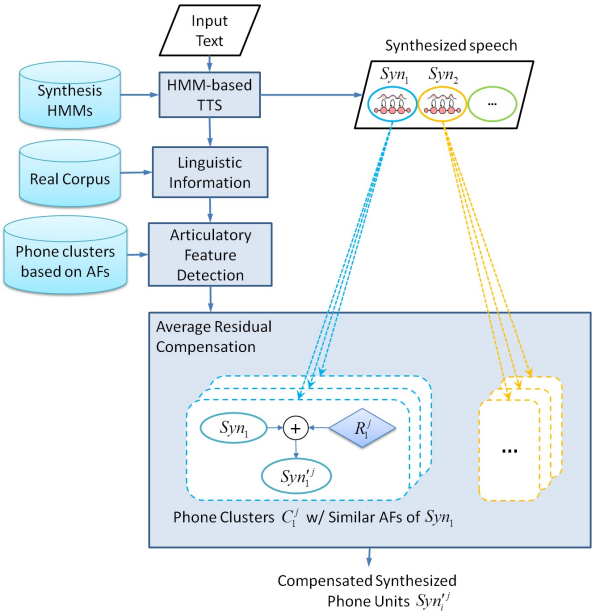


Figure 2: The average residual compensation of the proposed hybrid TTS system.

In the synthesis phase, the linguistic and the articulatory features of the input text and synthesized speech are extracted by a text analyzer and the pre-trained ANN, respectively. Here, the linguistic information used only consists of phone identity and phone duration for expanding the candidate units with similar linguistic information and articulatory features. After selecting some phone units in the natural corpus based on linguistic information for the i-th synthesized phone unit Syn_i, the articulatory features are extracted and then adopted to find suitable pre-trained phone clusters C_i^j, and $j = 1, ..., J$. The synthesis units of these selected phone clusters are potential candidate units for selection to synthesize the speech output. To select suitable units among them, the average residual model R_i^j of the cluster C_i^j is used to compensate the difference between the feature vectors of the HMM-based synthesized speech and the mean of phone units in the cluster.

The second part of the synthesis phase is to expand the candidate phone units based on the compensated feature vector of the synthesized unit since it becomes a more reliable criterion to select suitable natural phones for speech synthesis. The most similar k phone units u_i^k in the cluster are selected as the candidate phone units of natural speech for the following optimal phone sequence selection. The dynamic programming algorithm is applied to select the most suitable phone sequence of the target speech. In this study, there are three factors considered to calculate the cost in the dynamic programming algorithm: conventional cepstral distance, articulatory feature of nearby frames, and the pitch value. The distance measure based on unit-selection method is used as the selection criterion considering the three factors, which are described as follows:

$$C\left(t_1^n, u_1^n\right) = \sum_{i=1}^{n} C^t\left(t_i, u_i\right) + \sum_{i=2}^{n} C^c\left(u_{i-1}, u_i\right) \quad (2)$$

where $C^t\left(t_i, u_i\right)$ is the target cost between target unit t_i and candidate units u_i, and j is the index of the target phone unit. $C^c\left(u_{i-1}, u_i\right)$ is the concatenation cost between candidate unit u_{i-1} and candidate units u_i. The number of phone units of the target speech is n. In this study, the target cost considers both the linguistic information and articulatory features, which are used to cluster the phone units, so the concatenation cost is used to select the target phone units from these candidate phone units. The concatenation cost is defined as:

$$C^c\left(u_{i-1}, u_i\right) = w \cdot \tilde{d}_s\left(u_{i-1}, u_i\right) + (1-w) \cdot \tilde{d}_a\left(u_{i-1}, u_i\right) \quad (3)$$

where $\tilde{d}_s\left(u_{i-1}, u_i\right)$ is the normalized Euclidean cepstral distance between the boundary frames of the candidate phone unit u_{i-1} and u_i. $\tilde{d}_a\left(u_{i-1}, u_i\right)$ is the normalized articulatory feature of the nearby frames of the current unit. Normalization of the range to lie between 0 to 1 is performed using min-max normalization method. The original $d_a\left(u_{i-1}, u_i\right)$ is defined as follows:

$$d_a(u_{i-1}, u_i) = \frac{1}{2}\left(\left\|AF_i - AF_{i-1}^{+}\right\|^2 + \left\|AF_{i-1} - AF_i^{-}\right\|^2\right) \quad (4)$$

where AF_{i-1} is the articulatory feature of the last frame of the candidate phone unit u_{i-1}, while AF_{i-1}^{+} is the articulatory feature of the next frame of the last frame of phone unit u_{i-1} in its original speech utterance. Similarly, AF_i is the articulatory feature of the first frame of the candidate phone unit u_i, and AF_i^{-} is the articulatory feature of the previous frame of the first frame of phone unit u_i in its original speech utterance. The contextual information of the articulatory feature should be useful for selecting suitable unit in which their nearby frames have similar articulatory features. The linear combination weight w is adopted for combining these two distance measures.

Besides the two distance measures, the pitch value of the candidate unit is also considered because the prosodic information is an important factor for perceiving natural speech. The global intonations of prosodic patterns are calculated and clustered based on the number of syllables in the prosodic phrase and the position of the prosodic phrase in the utterance. When selecting the phone unit, its mean pitch value should not exceed the threshold of the global intonation of the prosodic phrase. Finally, the speech output for the input text is synthesized from the feature vectors of the selected phone units using the MLSA filter.

3. Evaluation

3.1. Speech Data Collection and Experimental Setup

The phonetically balanced sentences from the TsingHua-Corpus of Speech Synthesis (TH-CoSS) [15] were used for training the general HMM-based synthesis models [16]. The context-dependent phone labels are constructed automatically based on the linguistic features extracted from the database. For Mandarin speech, 107 phone units (including pause) were selected to construct a Mandarin HMM-based TTS system [17]. Each Mandarin syllable consists of two vowel phone units and one optional consonant phone unit. The sampling rate of the speech signals is 16kHz and the smoothed spectral coefficients were extracted using the STRAIGHT algorithm. A 5-state left-to-right HMM is adopted to model the acoustic features. The HMM-based Mandarin TTS system was trained using the speech data from a female speaker, including $5,406$ utterances with $98,749$ syllables. The HMMs are used to generate the same $5,406$ sentences for calculating the residual models of phone clusters.

For phone clustering based on articulatory features, a total of $5,406$ sentences uttered by the female speakers were used.

Phone unit clustering based on the phone identity and articulatory features is used for clustering all phones in the training corpus. After clustering, the residual model of each cluster is calculated by firstly aligning the synthesized phone unit to natural phone unit in the same utterances, and linear interpolation is used to make sure that each state of phone unit in the same cluster has the same length to calculate the average residual model. The differences between natural phone and synthesized phone are calculated and the linear regression of the average differences of each state are then calculated. Finally, the average residual models can be obtained for each phone cluster.

In order to evaluate the performance of the proposed hybrid Mandarin TTS system, experiments were conducted based on three different systems:

1. Conventional HMM-based TTS system [17];
2. Conventional Hybrid TTS system based on linguistic information and spectral distance;
3. Proposed Hybrid TTS system with residual compensation.

The only difference between the second and the third systems is the use of the average residual model.

3.2. Subjective Evaluations of Speech Quality and Naturalness

For subjective evaluation of speech quality, 5-point Mean Opinion Score (MOS) subjective evaluation was conducted to evaluate the effectiveness of the proposed method and the methods for comparison.

There were 20 speech utterances, selected from the daily newspapers, synthesized by the proposed method and the two methods for comparison. Ten native Mandarin subjects were invited to participate in the evaluation. The participants were asked to score the speech quality for each utterance.

The ABX preference test was also conducted to compare the naturalness of the synthesized speech. Subjects were presented some natural utterances of the target speaker as the references of natural speech, and then two synthesized utterances of the comparing methods were presented. Subjects were asked to select which synthesized speech is more natural.

The results of MOS evaluation are shown in Fig. 3. The proposed method achieved better scores compared to that of the HMM-based method significantly (p-value is 0.0013), and has similar performance to the conventional hybrid system. From the feedback of subjects, the speech quality and dynamic range of the two hybrid-based systems is much better than the HMM-based system. Fig. 4 shows the results of two ABX evaluations: one is to compare the proposed system to the HMM-based system, and the other is to compare the proposed system to the conventional hybrid system. The first comparison shows that the naturalness of the proposed system is much higher than the HMM-based method, and the generated speech is much similar to the natural speech. In the second comparison, the preference score between two hybrid systems is not statistically significant (p-value is 0.079), which means the performance of naturalness of both systems is similar. However, there are discontinuities in some utterances generated from the conventional hybrid system, while the proposed method rarely happens. From the analysis of the phenomenon, the proposed method alleviates the discontinuity problem by the expanded candidate units, which have better articulatory features or pitch values for concatenation than the conventional hybrid system.

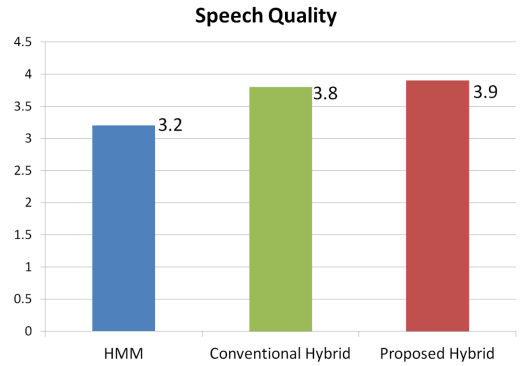


Figure 3: Speech quality comparison results of MOS test.

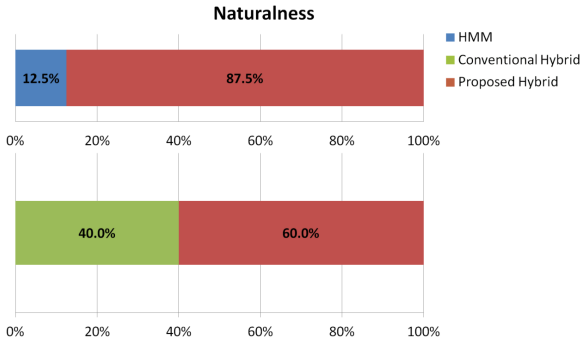


Figure 4: ABX test result of speech naturalness.

4. Conclusions

In this study, a hybrid method, which combines the unit-selection and the HMM-based method, is proposed to compensate the residuals between the feature vectors of the natural phone units and the HMM-synthesized phone units. Articulatory features are adopted to cluster the phone units with similar articulation to construct the residual models of phone clusters. The average residual model is modeled by the state-level linear regression models. With the residual model, the difference between natural speech and synthesized speech can be alleviated, and can be more reliable to select suitable phone units for speech synthesis with larger dynamic range and higher speech quality. The contextual information of articulatory feature is also helpful for selecting an optimal phone sequence from the candidate phone units. The results of subjective evaluation show that the proposed method achieved better performance than the HMM-based synthesizer with respect to the speech quality and naturalness. Comparing with the conventional hybrid system, the proposed system has similar performance of speech quality and naturalness. However, discontinuities of the synthesized speech for the conventional hybrid-based system are worse than that for the proposed system.

Future work will be focused on constructing a more precise residual model to compensate the residual and considering dynamic features of the constructed feature vectors for further improvement of speech quality and naturalness.

5. References

[1] A. Black, H. Zen, and K. Tokuda, "Statistical parametric speech synthesis," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, vol. 4, april 2007, pp. IV–1229 –IV–1232.

[2] C.-H. Wu and J.-H. Chen, "Automatic generation of synthesis units and prosodic information for chinese concatenative synthesis," *Speech Communication*, vol. 35, no. 3 - 4, pp. 219 – 237, 2001.

[3] K. Tokuda, H. Zen, and A. Black, "An hmm-based speech synthesis system applied to english," in *Proceedings of 2002 IEEE Workshop on Speech Synthesis*, September 2002, pp. 227 – 230.

[4] T. Anastasakos, J. McDonough, R. Schwartz, and J. Makhoul, "A compact model for speaker-adaptive training," in *Proceedings of Fourth International Conference on Spoken Language*, vol. 2, oct 1996, pp. 1137 –1140 vol.2.

[5] Y. Qian, J. Xu, and F. Soong, "A frame mapping based HMM approach to cross-lingual voice transformation," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, May 2011, pp. 5120 –5123.

[6] C.-P. Chen, Y.-C. Huang, C.-H. Wu, and K.-D. Lee, "Cross-lingual frame selection method for polyglot speech synthesis," in *Proceedings of International Conference on Acoustics, Speech and Signal Processing*, march 2012, pp. 4521 –4524.

[7] C.-C. Hsia, C.-H. Wu, and J.-Q. Wu, "Conversion function clustering and selection using linguistic and spectral information for emotional voice conversion," *IEEE Transactions on Computers*, vol. 56, no. 9, pp. 1245 –1254, sept. 2007.

[8] H. Kawai, T. Toda, J. Ni, M. Tsuzaki, and K. Tokuda, "Ximera: A new tts from atr based on corpus-based technologies," in *Proceedings of the Fifth ISCA Workshop on Speech Synthesis (SSW5)*, 2004, pp. 179–184.

[9] J. hui Yang, Z. wei Zhao, Y. Jiang, G. ping Hu, and X. ru Wu, "Multitier non-uniform unit selection for corpus-based speech synthesis," in *Blizzard Challenge Workshop*, 2006.

[10] S. Krstulović, J. Latorre, and S. Buchholz, "Comparing QMT1 and HMMs for the synthesis of American English prosody," in *Proceedings of Speech Prosody*, 2008.

[11] Z.-H. Ling and R.-H. Wang, "Minimum unit selection error training for hmm-based unit selection speech synthesis system," in *IEEE International Conference on Acoustics, Speech and Signal Processing, 2008. ICASSP 2008.*, 2008, pp. 3949–3952.

[12] M. Ostendorf, "Moving beyond the 'beads-on-a-string' model of speech," in *Proceedings of IEEE ASRU Workshop.*, 1999, pp. 79–84.

[13] S. Stuker, F. Metze, T. Schultz, and A. Waibel, "Integrating multilingual articulatory features into speech recognition," in *Proceedings of Eurospeech*, 2003, pp. 1033–1036.

[14] C. Ma, Y. Tsao, and C.-H. Lee, "A study on detection based automatic speech recognition." in *Proceedings of Interspeech 2006*, 2006.

[15] L. Cai, D. Cui, and R. Cai, "TH-CoSS, a mandarin speech corpus for tts," *Journal of Chinese Information Processing*, vol. 21, no. 2, pp. 94–99, 2007.

[16] H. Zen, T. Nose, J. Yamagishi, S. Sako, T. Masuko, A. W. Black, and K. Tokuda, "The HMM-based speech synthesis system version 2.0," in *Proceedings of ISCA SSW6*, August 2007.

[17] C.-C. Hsia, C.-H. Wu, and J.-Y. Wu, "Exploiting prosody hierarchy and dynamic features for pitch modeling and generation in hmm-based speech synthesis," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 8, pp. 1994 –2003, nov. 2010.

Author Index